Bayesian Semi-Parametric Approaches to Exposure-Response Modeling with Time-to-Event Outcomes

The 79th Annual Deming Conference on Applied Statistics
Philadelphia, PA

Tim Waterhouse

4 December 2023

METRUM
RESEARCH GROUP

Table of Contents

Outline

Bayesian Semi-Parametric Approaches to Exposure-Response Modeling with Time-to-Event Outcomes

- Introduction to exposure-response (E-R) concepts

Outline

Bayesian Semi-Parametric Approaches to Exposure-Response Modeling with Time-to-Event Outcomes

- Introduction to exposure-response (E-R) concepts
- Introduction to survival models (from an E-R perspective)

Outline

Bayesian Semi-Parametric Approaches to Exposure-Response Modeling with Time-to-Event Outcomes

- Introduction to exposure-response (E-R) concepts
- Introduction to survival models (from an E-R perspective)
- Parametric and semi-parametric E-R survival models

Outline

Bayesian Semi-Parametric Approaches to Exposure-Response Modeling with Time-to-Event Outcomes

- Introduction to exposure-response (E-R) concepts
- Introduction to survival models (from an E-R perspective)
- Parametric and semi-parametric E-R survival models
- Bayesian estimation of parametric E-R survival models

Outline

Bayesian Semi-Parametric Approaches to Exposure-Response Modeling with Time-to-Event Outcomes

- Introduction to exposure-response (E-R) concepts
- Introduction to survival models (from an E-R perspective)
- Parametric and semi-parametric E-R survival models
- Bayesian estimation of parametric E-R survival models
- Bayesian estimation of semi-parametric E-R survival models

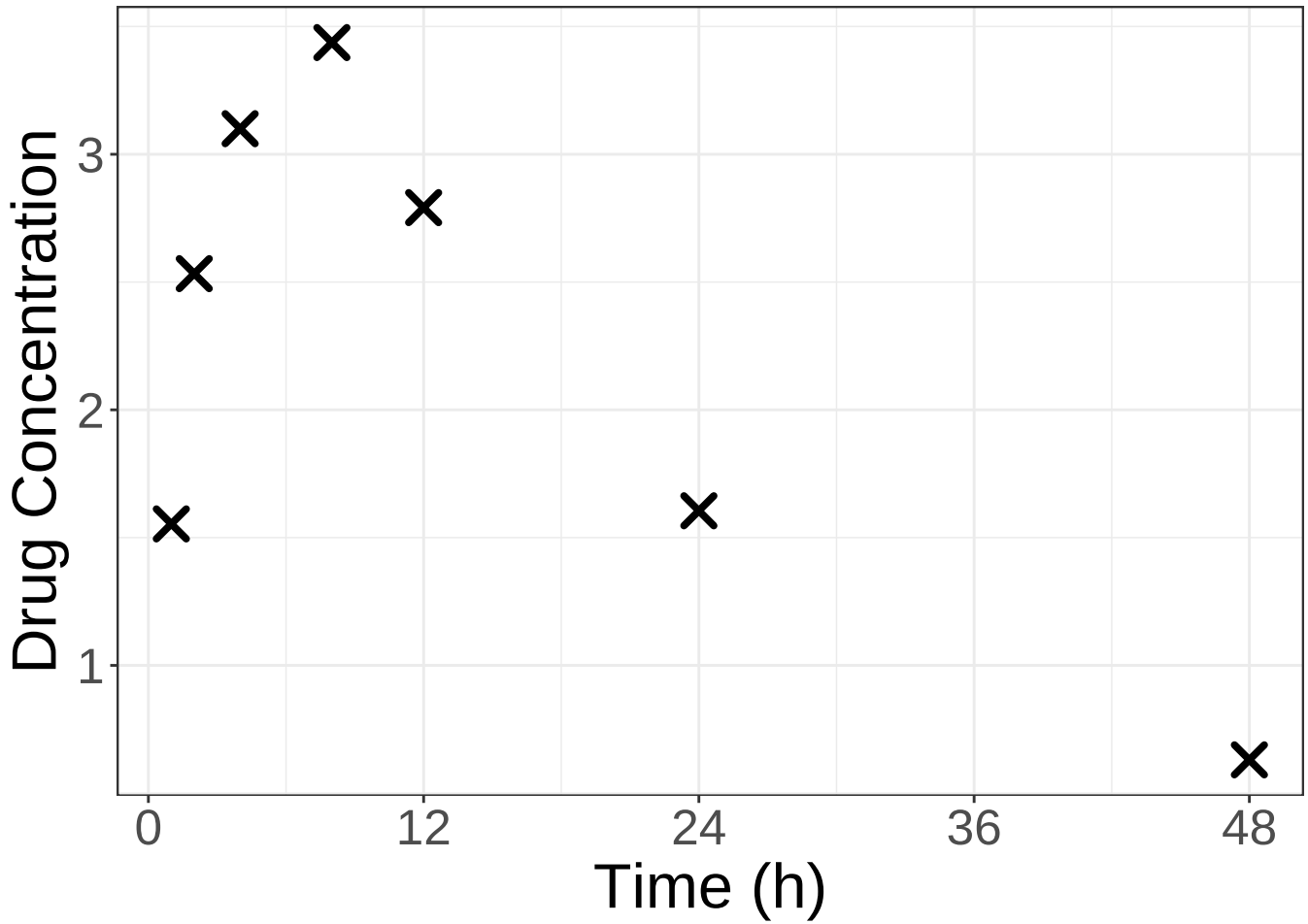
Table of Contents

What do we mean by exposure?

- Any measure of how much drug a person is exposed to:
  - Dose (e.g., daily dose, total dose)
  - Drug concentration in the body at some time point (e.g., maximum concentration following the first dose, average concentration during the dosing interval at steady state)
  - Time above some threshold (e.g., time above minimum effective concentration)
  - Integrated measure of concentration (e.g., area under the concentration-time curve at steady state)

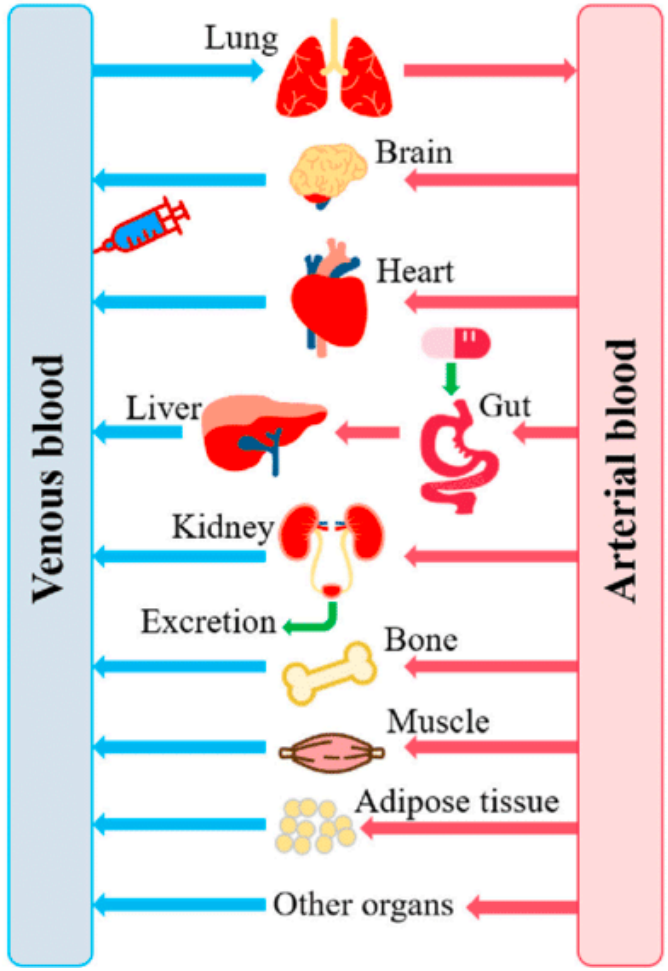
(Ruiz-Garcia et al. 2023)

Observed concentration-time data



Physiologically based pharmacokinetic model (PBPK)

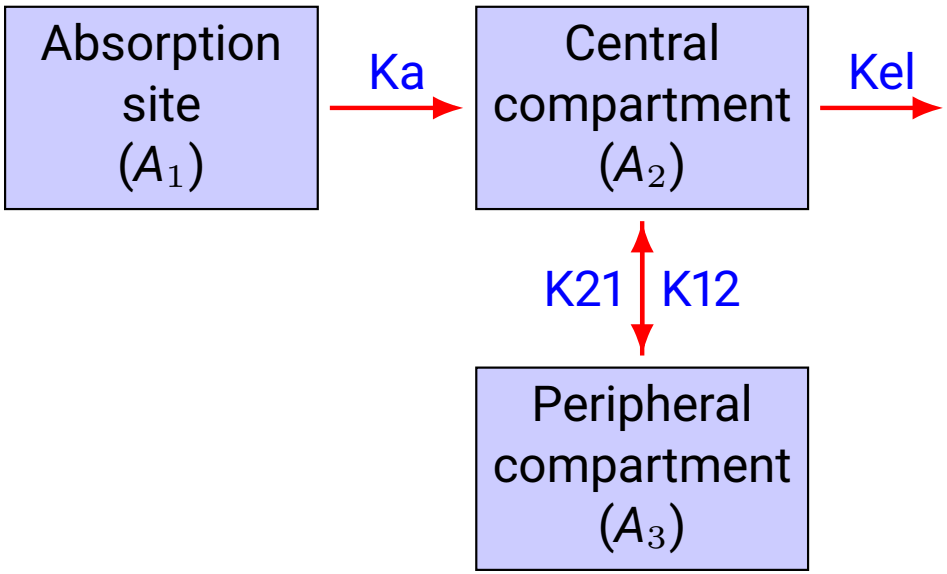
This data can be modeled very mechanistically …



Peng, Cheng, and Xie (2021)

Compartmental pharmacokinetic (PK) model

… or not so mechanistically

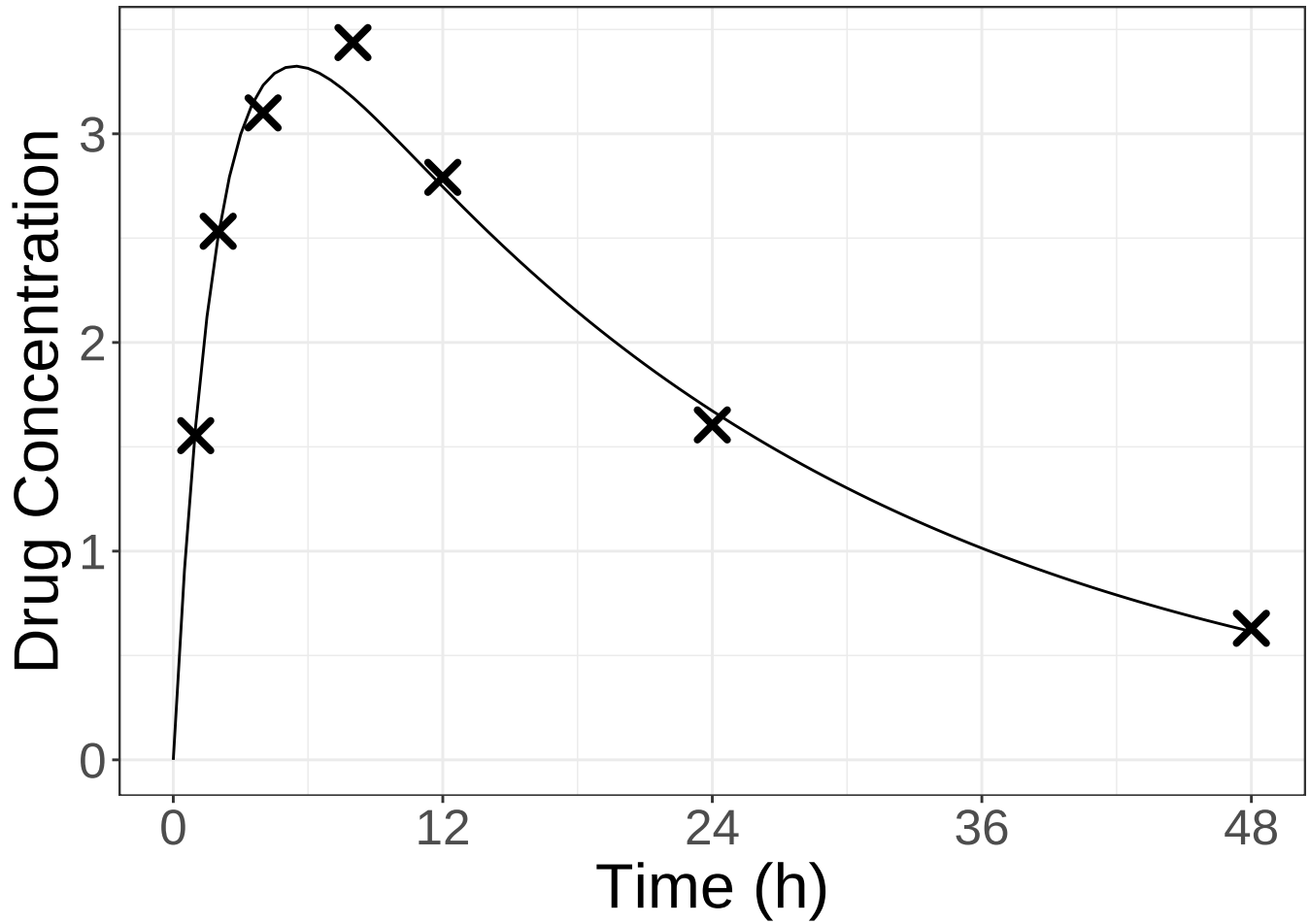


$$\frac{dA_1}{dt} = -Ka \times A_1$$

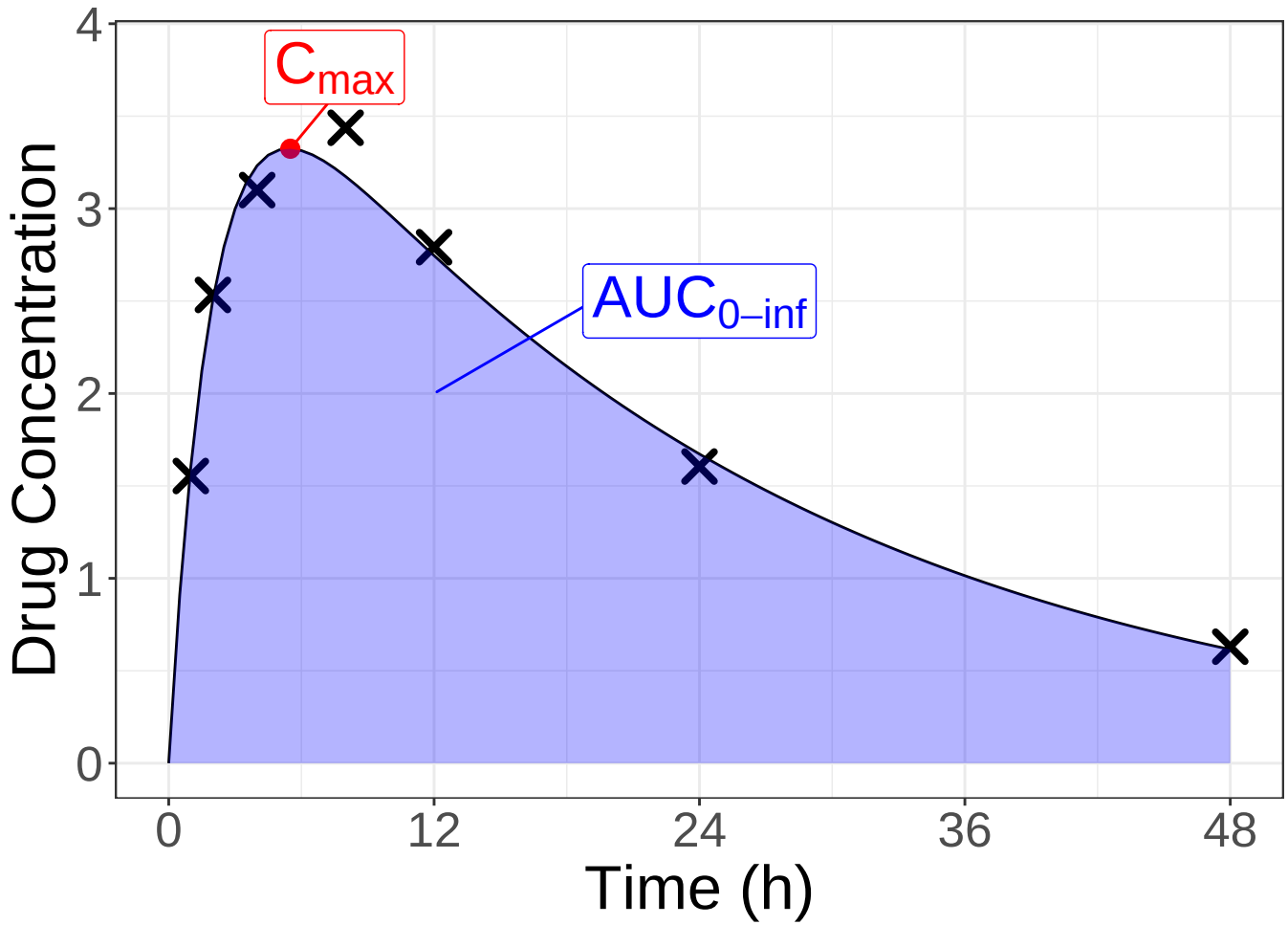
$$\frac{dA_2}{dt} = Ka \times A_1 - (K12 + Kel) \times A_2 + K21 \times A_3$$

$$\frac{dA_3}{dt} = K12 \times A_2 - K21 \times A_3$$

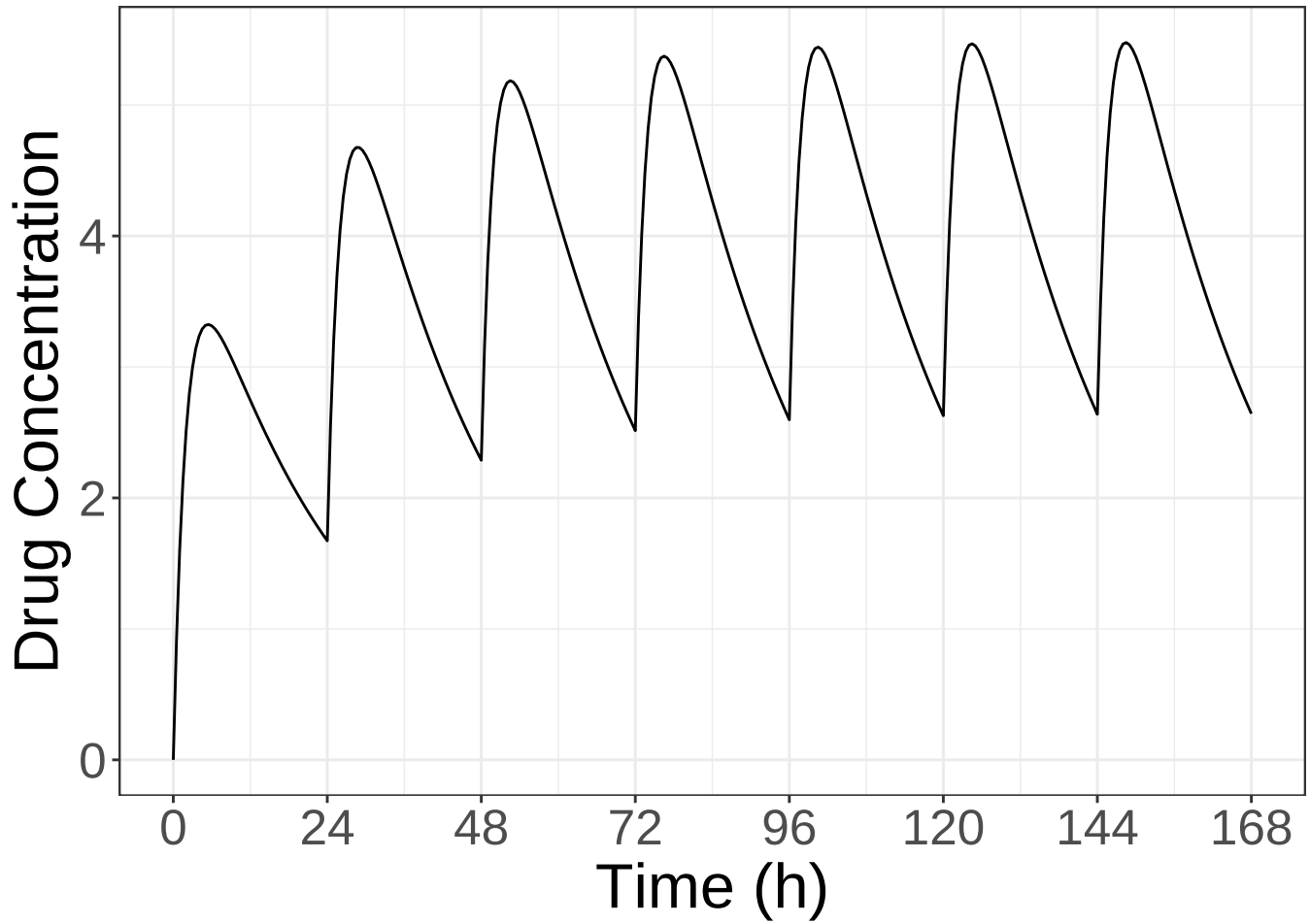
Simulation from fitted PK model



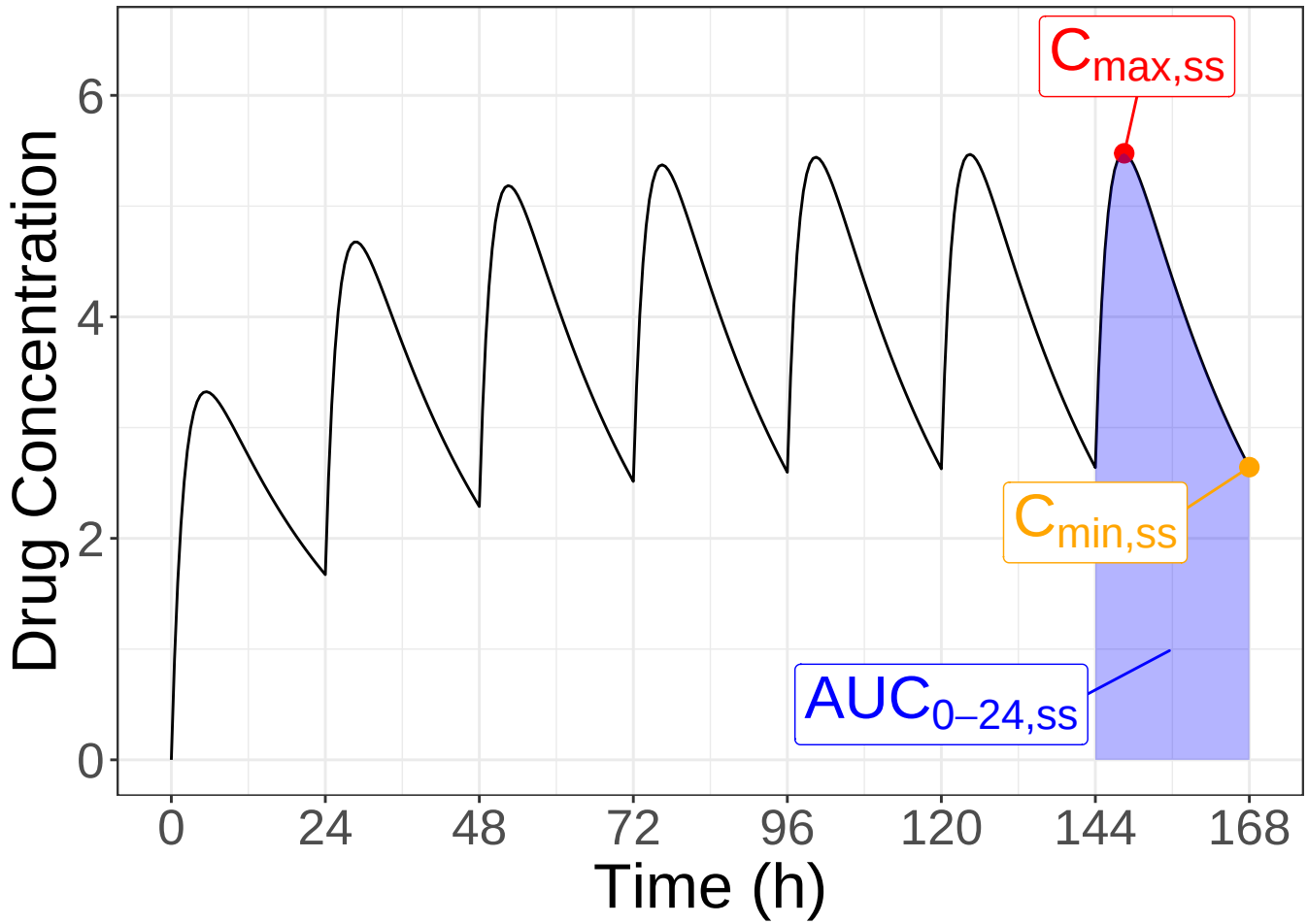
Single dose exposure metrics



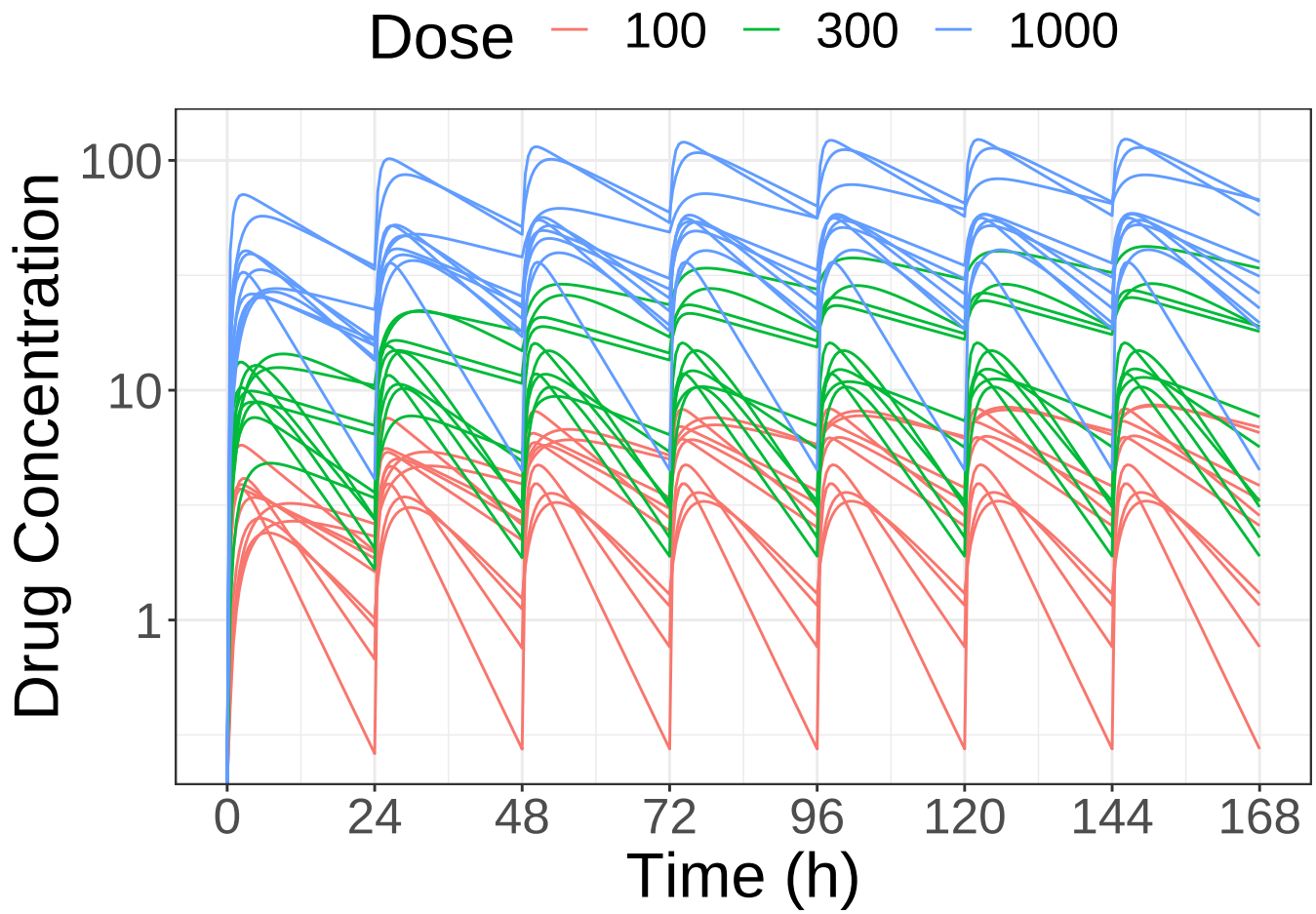
Multiple dose PK simulation



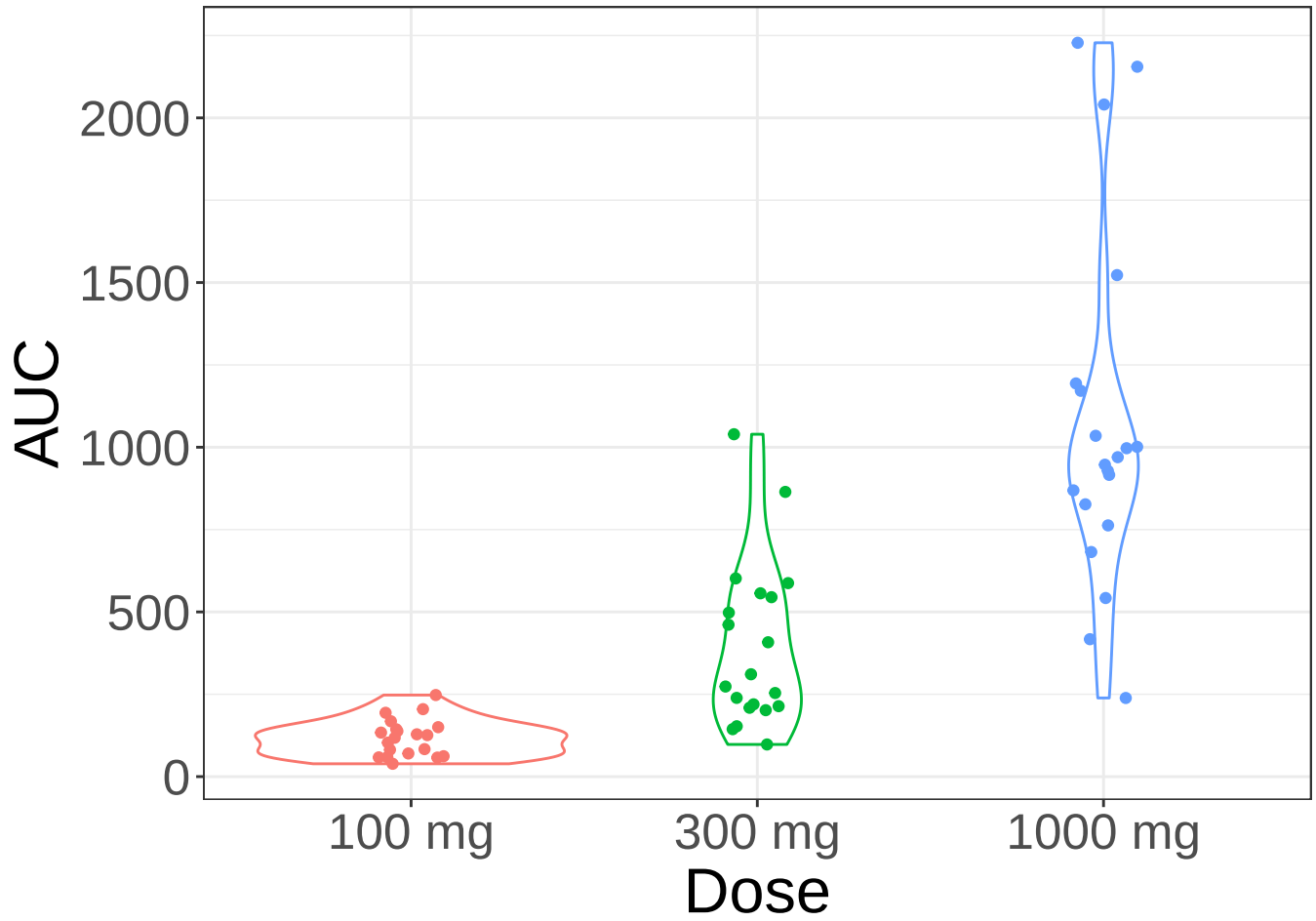
Multiple dose exposure metrics



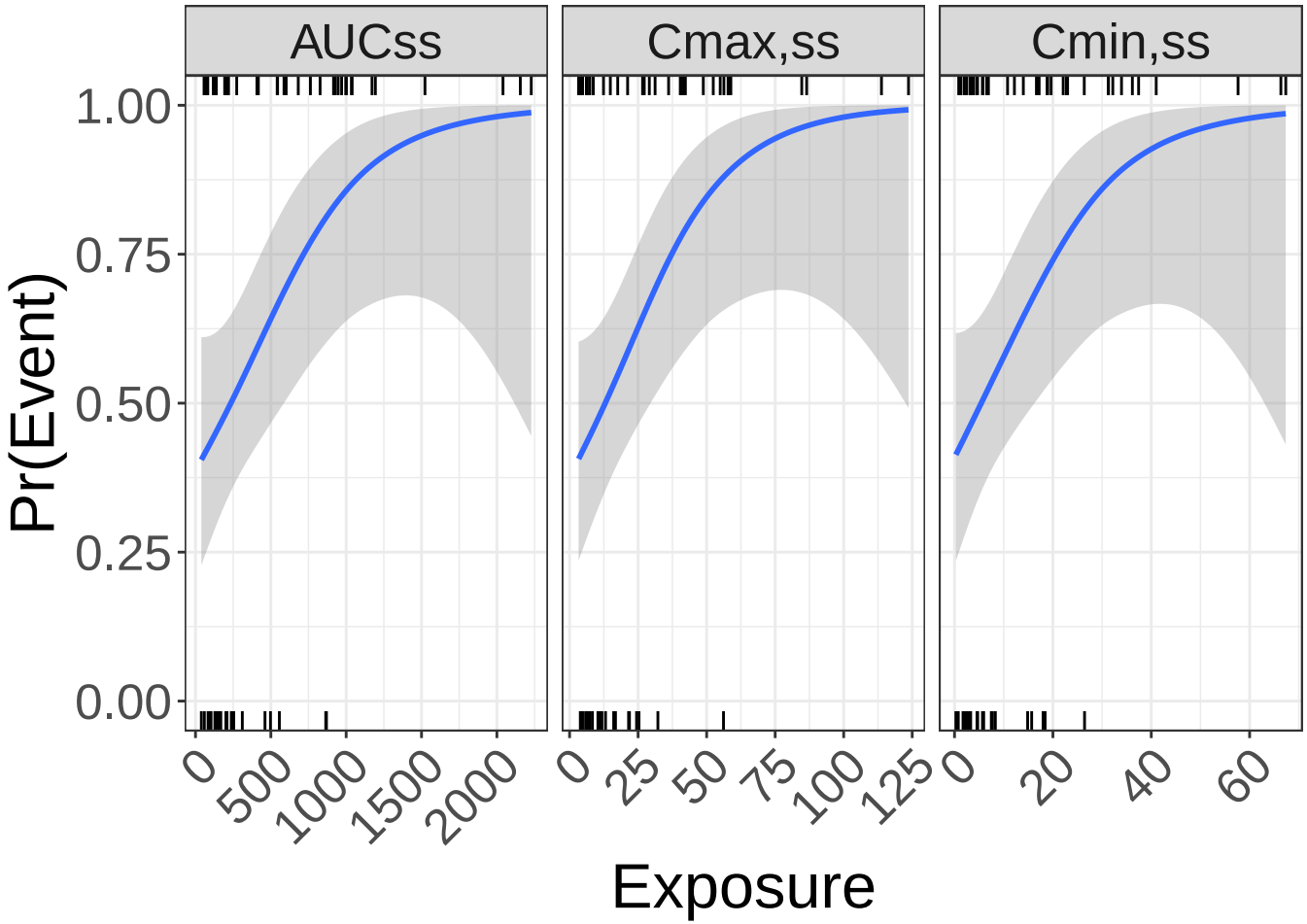
Multiple dose PK simulation with between-subject variability



Dose vs exposure



Exposure-response for binary endpoint



Exposure-response: Logistic regression model

We fit a basic logistic regression model:

$$\eta = \beta_0 + \beta_1 AUC$$

$$P(Y = 1|AUC) = \frac{e^{\eta}}{1 + e^{\eta}}$$

The model is fit in `brms` with default (flat) priors (not something we'd typically do in practice)

Fitted logistic regression model

```
##  Family: bernoulli
##   Links: mu = logit
## Formula: event ~ AUC
##    Data: sim_auc (Number of observations: 60)
##   Draws: 4 chains, each with iter = 2000; warmup = 500; thin = 1;
##          total post-warmup draws = 6000
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept     0.78      0.35     0.15     1.50 1.00     2395     2193
## AUC           1.47      0.50     0.58     2.56 1.00     2643     2530
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potenti
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

Fitted logistic regression model: posterior predictive check

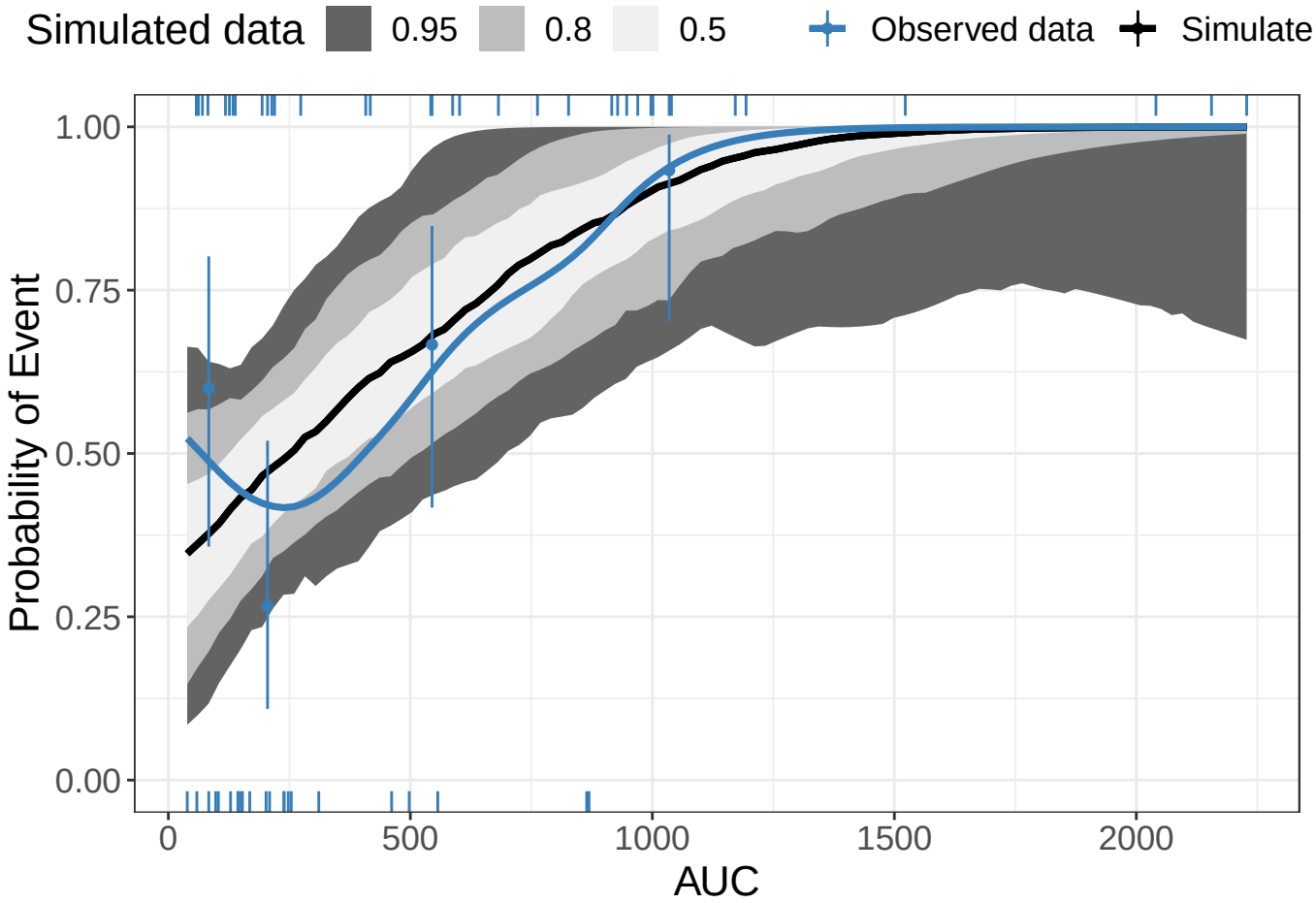


Table of Contents

What is Survival Analysis?

- Statistical method to analyze time-to-event data
- Commonly used in medical research and social sciences
- Deals with predicting the time until an event occurs

What is time-to-event data?

- In clinical studies, we often measure the time to a specific event:
  - time to death
  - time to disease worsening
  - time to incident adverse event
  - time to abnormal lab value (e.g., AST $> 3 \times$ ULN)
  - time to infection
  - time to study discontinuation
  - duration of hospital visit

Three essential components

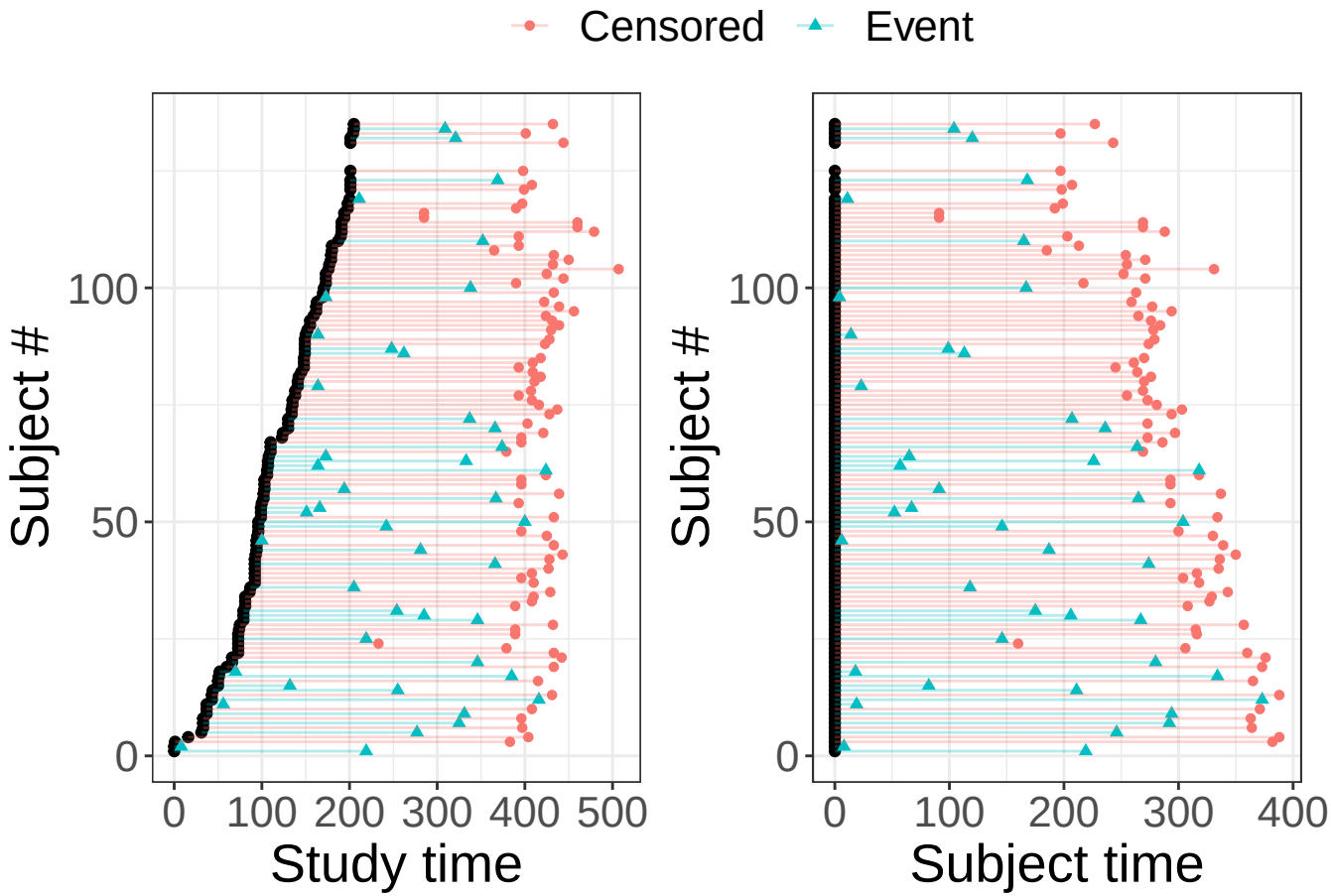
- Well-defined event
- Clear time origin
- Defined time scale

Quiz

What might be an event definition and time origin for *time to disease worsening* in a clinical trial?

What makes TTE data different?

- For some subjects, we may not observe an event
  - The time to event is **censored**



A little notation

- There are two time-to-event processes happening:
  - T = time to event of interest
  - C = time to censoring
- With right censoring, we observe
  - $T^* = \min(T, C)$
  - $\delta = I(T \leq C)$
- We are trying to estimate the distribution of T, but we observe T^*
  - We'll return to this when discussing model diagnostics
- Typical to assume that T and C are independent

Types of censoring

- Right censoring
  - We know the event did not happen prior to time b (i.e., we know $T > b$)
- Left censoring
  - We know the event happened before time a (i.e., we know $T < a$)
- Interval censoring
  - We know the event happened between times a and b but not the exact time (i.e, $a < T < b$)
- In clinical trials, we most often deal with right and interval censoring

How does censoring introduce complexity?

- If we observed event times for all subjects, we could use 'standard' methods
- Hard to estimate probability density function when we don't see all events happening
  - A type of missing data problem
- Working with the hazard function alleviates some of the problems
- Hazard function = instantaneous event rate, conditional on event happening on or after time t
  - $h(t) = \lim_{\Delta t \to 0} \frac{P(t < T \leq t + \Delta t \mid T \geq t)}{\Delta t}$
  - "conditional on event happening on or after time t" is what helps

Terminology

- Cumulative hazard = total hazard accumulated to time t
  - $H(t) = \int_0^t h(s)ds$
  - This is the expected number of events to time t (Hosmer, Lemeshow, and May 2011b)
- Probability density function = instantaneous event risk (aka *density*)
  - $f(t) = \lim_{\Delta t \to 0} \frac{P(t<T\leq t+\Delta t)}{\Delta t}$
- Survival function = probability of an event happening after time t
  - $S(t) = P(T > t)$

Connections

Two important relationships to remember:

- The relation between the survival function and the cumulative hazard

$$S(t) = \exp\{-H(t)\}$$

- We can derive the density function from the hazard and survival functions

$$f(t) = h(t)S(t)$$

Illustrating relationships between functions

Some examples to illustrate the relationships between the hazard, cumulative hazard, density, and survival functions for some parametric distributions.

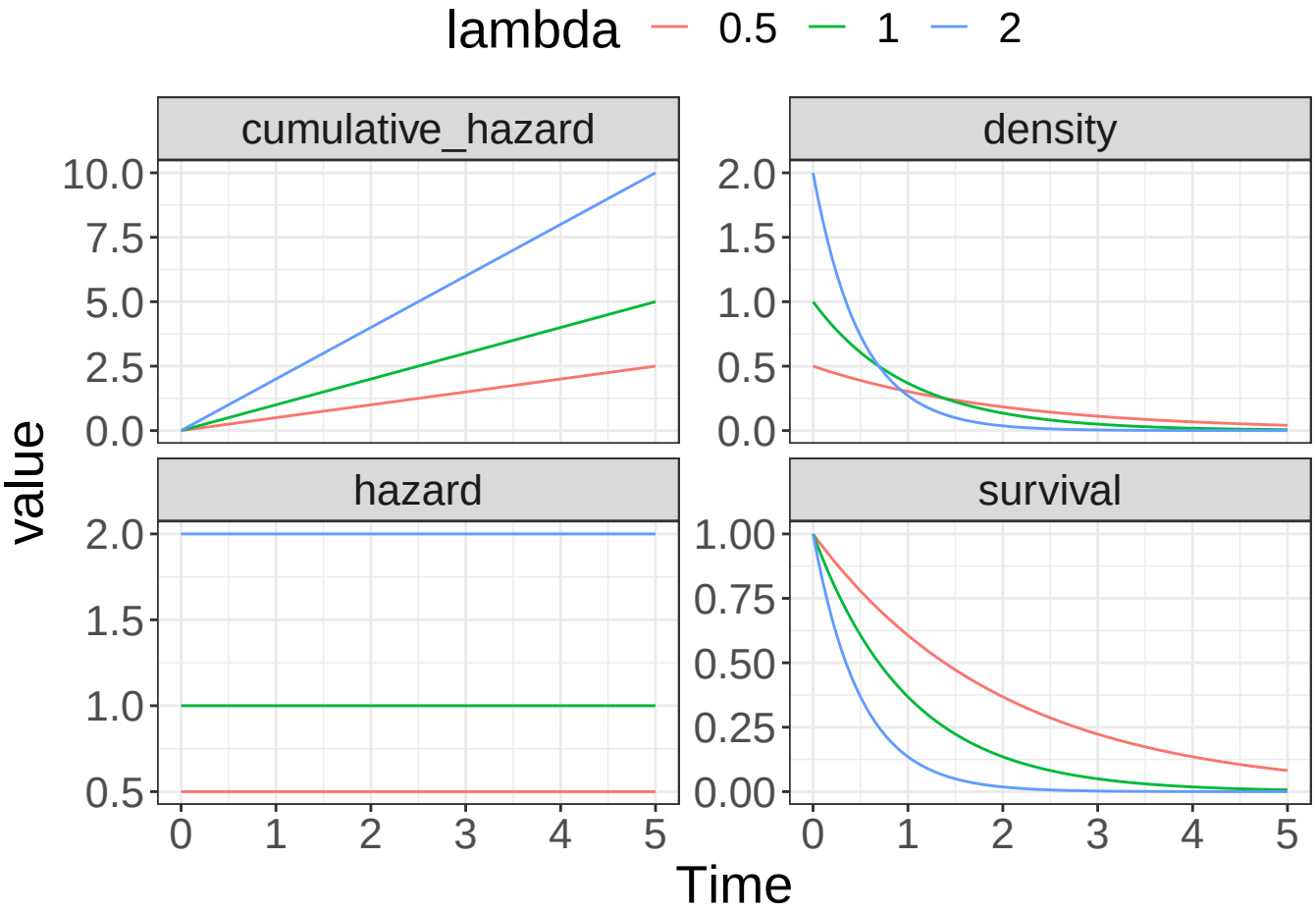
Exponential distribution

The hazard is constant as a function of time: $h(t) = \lambda$.

From first principles:

- h(t) = λ
- H(t) = λ t
- S(t) = exp(-λ t)
- f(t) = h(t) $\cdot$ S(t) = λ exp(-λ t)

Exponential distribution



Weibull distribution

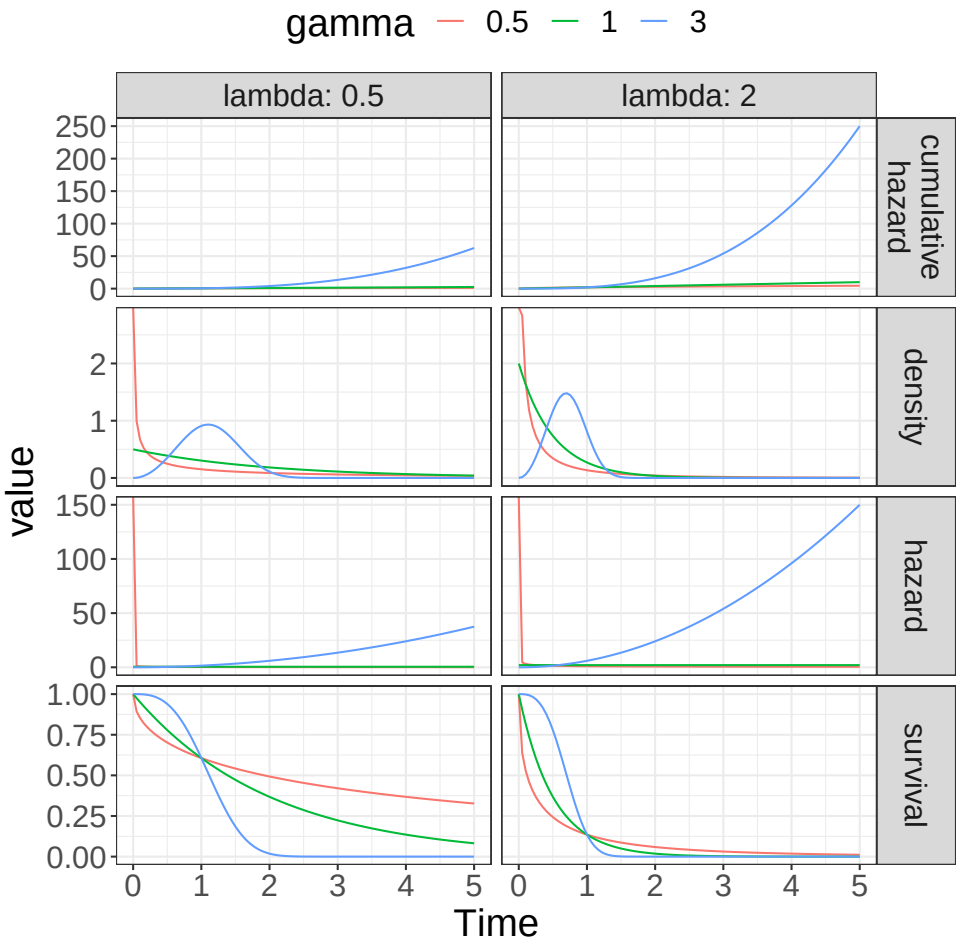
The Weibull distribution has two parameters: lambda and gamma
The log hazard is linear in the log of time:

$$h(t) = \gamma\lambda t^{\gamma-1} \iff \log h(t) = \log\gamma + \log\lambda + (\gamma-1)\log t$$

From first principles:

- h(t) = $\lambda\gamma t^{\gamma-1}$
- H(t) = λt^{γ}
- S(t) = $\exp(-\lambda t^{\gamma})$
- f(t) = $h(t) \cdot S(t) = \lambda\gamma t^{\gamma-1}\exp(-\lambda t^{\gamma})$

Weibull distribution



Non-parametric estimation of survival, cumulative hazard and hazard functions

- We'll start with non-parametric estimates of $S(t)$ and $H(t)$
- The most commonly used estimator of $S(t)$ is the Kaplan-Meier estimator
  - aka the ***product limit*** estimate
- The most common estimator of $H(t)$ is the `Nelson-Aalen` estimator
  - Can also estimate $S(t)$ as $\widehat{S_{FH}(t)} = \exp\left\{\widehat{H_{NA}(t)}\right\}$
  - This is known as the Fleming-Harrington estimate of $S(t)$
  - Similar, but not identical, to K-M estimate

Basics of Kaplan-Meier estimate

Suppose we have these 10 event times (in days):

```
##  [1] "81"   "81"   "88"   "88"   "92+"  "92"   "93"   "95"   "95"   "105+"
```

where a "+" denotes a censored observation.

How would you estimate

- $P(T > 80)$?

Basics of Kaplan-Meier estimate

Suppose we have these 10 event times (in days):

```
##  [1] "81"   "81"   "88"   "88"   "92+"  "92"   "93"   "95"   "95"   "105+"
```

where a "+" denotes a censored observation.

How would you estimate

- $P(T > 80)$?
- $P(T > 80) = 1$ because all event times are after 80 days

Basics of Kaplan-Meier estimate

Suppose we have these 10 event times (in days):

```
##  [1] "81"   "81"   "88"   "88"   "92+"  "92"   "93"   "95"   "95"   "105+"
```

where a "+" denotes a censored observation.

How would you estimate

- $P(T > 80)$?
- $P(T > 80) = 1$ because all event times are after 80 days
- $P(T > 90)$?

Basics of Kaplan-Meier estimate

Suppose we have these 10 event times (in days):

```
##  [1] "81"   "81"   "88"   "88"   "92+"  "92"   "93"   "95"   "95"   "105+"
```

where a "+" denotes a censored observation.

How would you estimate

- $P(T > 80)$?
- $P(T > 80) = 1$ because all event times are after 80 days

- $P(T > 90)$?
- $P(T > 90) = 6/10$ because we know exactly 4 events happened before 90 days

Basics of Kaplan-Meier estimate

Suppose we have these 10 event times (in days):

```
##  [1] "81"   "81"   "88"   "88"   "92+"  "92"   "93"   "95"   "95"   "105+"
```

where a "+" denotes a censored observation.

How would you estimate

- $P(T > 80)$?
- $P(T > 80) = 1$ because all event times are after 80 days
- $P(T > 90)$?
- $P(T > 90) = 6/10$ because we know exactly 4 events happened before 90 days
- $P(T > 94)$?

Basics of Kaplan-Meier estimate

Suppose we have these 10 event times (in days):

```
##  [1] "81"   "81"   "88"   "88"   "92+"  "92"   "93"   "95"   "95"   "105+"
```

where a "+" denotes a censored observation.

How would you estimate

- $P(T > 80)$?
- $P(T > 80) = 1$ because all event times are after 80 days

- $P(T > 90)$?
- $P(T > 90) = 6/10$ because we know exactly 4 events happened before 90 days

- $P(T > 94)$?
- $P(T > 94) =?$ We know 3 events happened after 94 days, but what about the censored time at 92 days?

Kaplan-Meier and conditional probability

It turns outs that we can use some basic probability calculations to estimate $S(t)$ in the presence of censoring.

1. Divide time into distinct intervals (at each event time, τ_j)
2. For each interval j,
   - Calculate the proportion of subjects with an event (d_j), among the subjects in the risk set for that interval (r_j)
   - The risk set at time t = N
     - number of events prior to t
     - number of censored prior to t
   - Calculate the probability of an event after the j^{th} interval, conditional on no event prior to the interval as $1 - \frac{d_j}{r_j}$
3. Estimate $S(t)$ as the product of the conditional probabilities up to time t
   - $\hat{S}_{KM}(t) = \prod_{\tau_j \leq t} \left(1 - \frac{d_j}{r_j}\right)$

Kaplan-Meier estimation in R

Fortunately, we don't have to do that work by hand :)

The `survfit` function in the R package `survival` does the work for us:

```
fit0 <- survfit(Surv(TTE, AE_any) ~ 1, data = dat_use)
```

Input data structure

```
## Rows: 180
## Columns: 15
## Groups: PBO [2]
## $ STUDYID    <fct> PROTA, PROTA, PROTA, PROTA, PROTA, PROTA, PROTA, PROT
## $ SEXTXT     <fct> MALE, MALE, MALE, MALE, MALE, MALE, MALE, MALE, MALE,
## $ PTTYPE     <fct> HV, HV, HV, HV, HV, HV, HV, HV, HV, HV, HV, HV, HV, H
## $ USUBJID    <fct> UID-001, UID-002, UID-003, UID-004, UID-005, UID-006,
## $ PBO        <chr> "PBO", "PBO", "PBO", "TRT", "TRT", "TRT", "PBO", "TRT
## $ CAVGSS     <dbl> 0.0000000, 0.0000000, 0.0000000, 1.9847466, 1.3657863
## $ BWT        <dbl> 77.36278, 82.95064, 83.85795, 77.99817, 87.71328, 77.
## $ STUDYDUR   <dbl> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
## $ AE01       <int> 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,
## $ AETOXGR    <fct> Mild, Mild, Mild, Mild, Mild, Severe, Severe, None, M
## $ TTE        <dbl> 0.035762378, 0.011968416, 0.902641754, 0.762268355, 0
## $ ae_any     <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, TRUE
## $ TTE_SEVERE <dbl> 2.00000000, 2.00000000, 2.00000000, 2.00000000, 2.000
## $ AE_any     <dbl> 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1,
## $ Quartile   <chr> "PBO", "PBO", "PBO", "Q4", "Q3", "Q3", "PBO", "Q3", "
```

Input data structure

The data includes exposure as continuous (CAVGSS) and discrete (Quartile) columns.

```
## # A tibble: 180 x 6
## # Groups:   PBO [2]
##    PBO   USUBJID    TTE AE_any CAVGSS Quartile
##    <chr> <fct>    <dbl>  <dbl>  <dbl> <chr>
##  1 PBO   UID-001 0.0358      1  0     PBO
##  2 PBO   UID-002 0.0120      1  0     PBO
##  3 PBO   UID-003 0.903       1  0     PBO
##  4 TRT   UID-004 0.762       1  1.98  Q4
##  5 TRT   UID-005 0.313       1  1.37  Q3
##  6 TRT   UID-006 0.0657      1  1.41  Q3
##  7 PBO   UID-007 0.0707      1  0     PBO
##  8 TRT   UID-008 2           0  1.21  Q3
##  9 TRT   UID-009 1.9         1  2.22  Q4
## 10 TRT   UID-010 1.9         1  0.383 Q1
## # i 170 more rows
```

Kaplan-Meier estimation in R

```
fit0 <- survfit(Surv(TTE, AE_any) ~ 1, data = dat_use)
```

- The `Surv(time, event)` function creates a survival response object
  - `time` = event or censoring time
  - `event` = event indicator (1=event, 0 = right censored)
  - More complex types of censoring can be handled
- RHS of formula cannot include continuous variables (*Why?*)
  - This is okay: `survfit(Surv(TTE, AE_any) ~ Quartile, data=dat_use)`
  - This is not: `survfit(Surv(TTE, AE_any) ~ CAVGSS, data=dat_use)`

Basic Survfit output

The `survfit` object gives us some basic information:

```
print(fit0)
```

```
## Call: survfit(formula = Surv(TTE, AE_any) ~ 1, data = dat_use)
##
##         n events median 0.95LCL 0.95UCL
## [1,] 180    132   0.32   0.139   0.649
```

More Survfit output

We can get more detail and predicted values with `summmary`

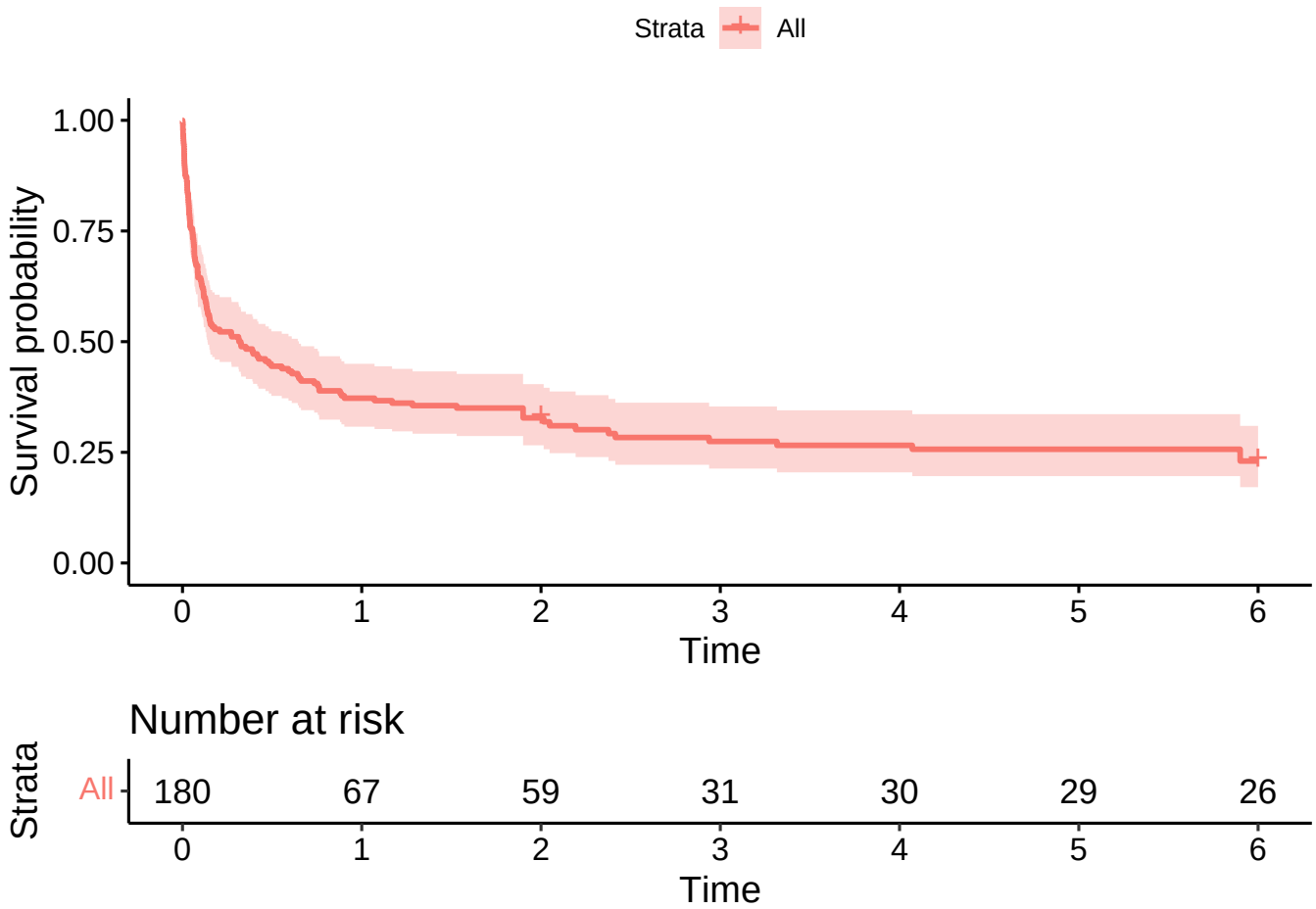
```
summary(fit0, times = seq(0, 1, by = 0.25))
```

```
## Call: survfit(formula = Surv(TTE, AE_any) ~ 1, data = dat_use)
##
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##  0.00    180       0    1.000  0.0000        1.000        1.000
##  0.25     94      86    0.522  0.0372        0.454        0.601
##  0.50     80      14    0.444  0.0370        0.377        0.523
##  0.75     73       7    0.406  0.0366        0.340        0.484
##  1.00     67       6    0.372  0.0360        0.308        0.450
```

Plotting the estimated survival function

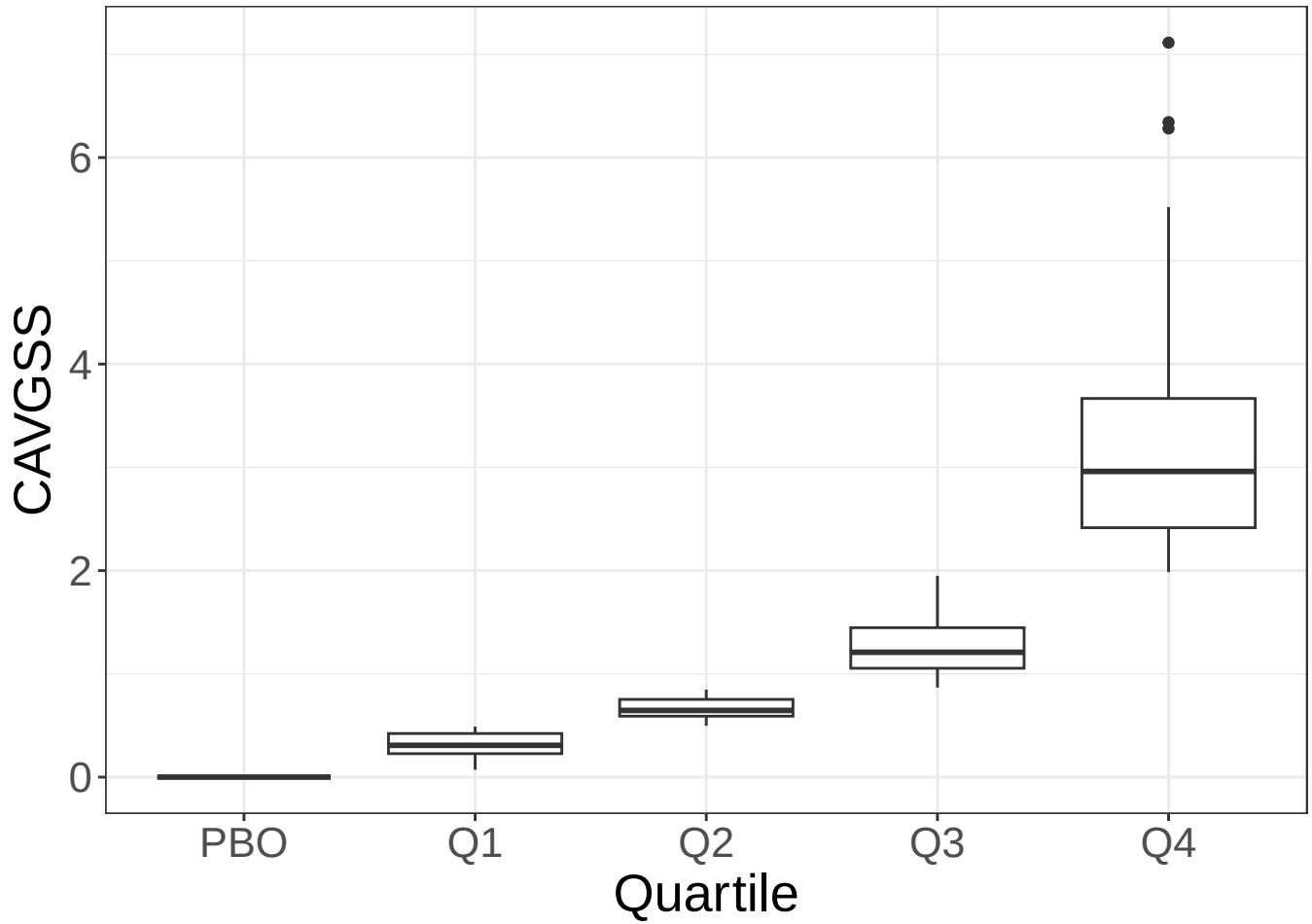
The `survminer::ggsurvplot` function provides clean plots

```
survminer::ggsurvplot(fit0, risk.table = TRUE, data = dat_use)
```

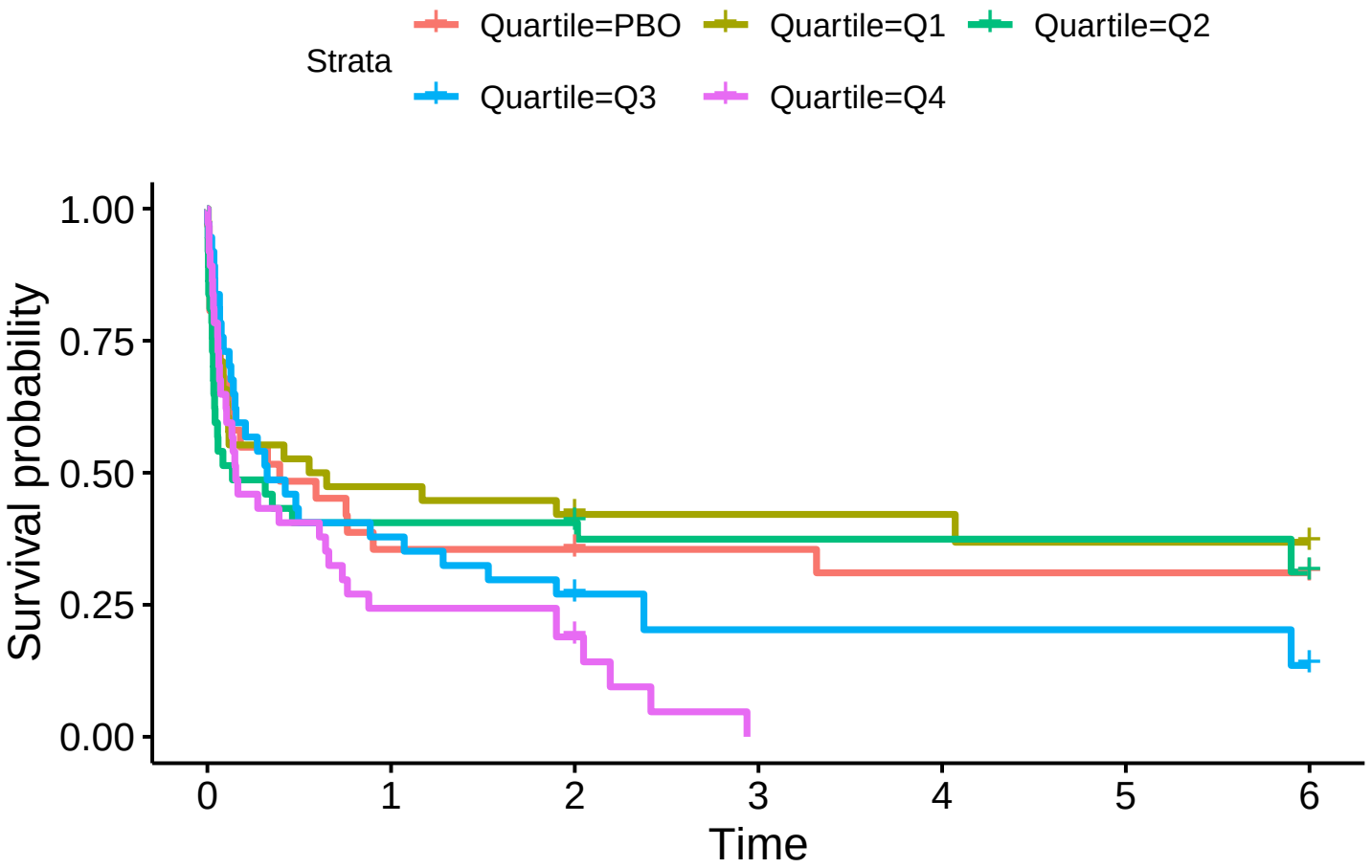


Let's explore effects of exposure on survival

First, what does our exposure data look like? CAVGSS is $C_{avg,ss}$, the average concentration over a dosing interval at steady state.

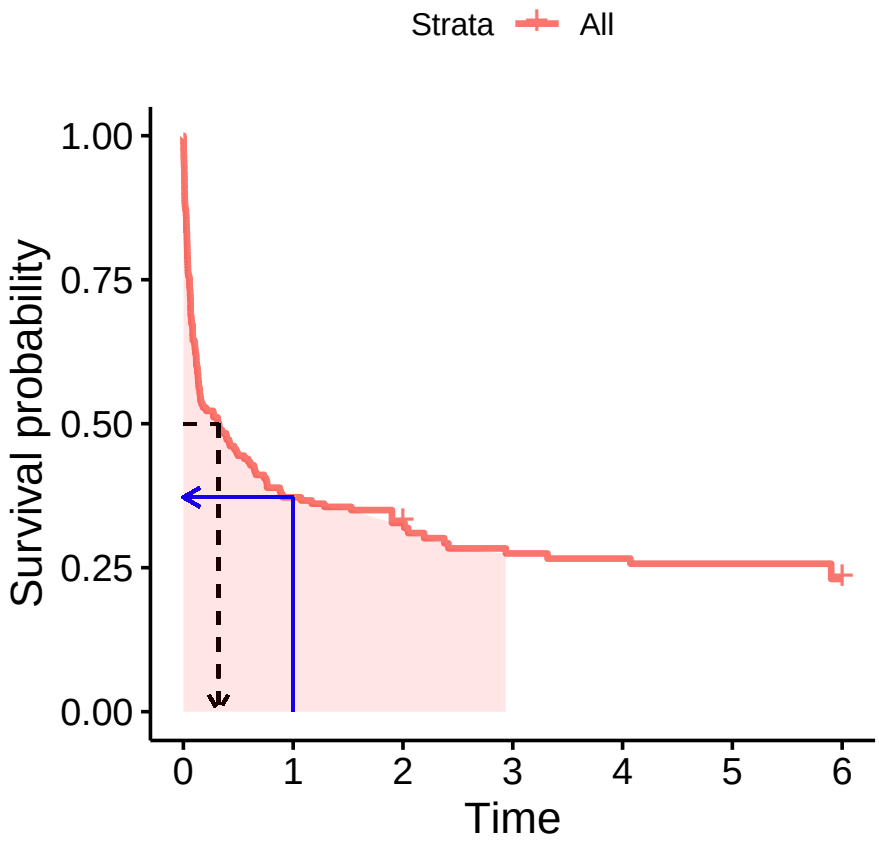


Exposure-response using exposure quartiles



Summary measures of S(t)

- Median time to event (black dashed line)
- Event rate at time *t* (blue dashed line)
- Restricted mean survival time (RMST) to t^*
  - (Unrestricted) mean survival may not be well-defined
  - RMST is the average event-free time up to t^*
  - Equivalent to the area under S(t) from 0 to t^*



Summary measures of S(t) in R: median

- Median time to event

```
quantile(fit0, probs = 0.50) %>% unlist()
```

```
## quantile.50     lower.50     upper.50
##   0.3200023    0.1393874    0.6492904
```

Summary measures of S(t) in R: percentile

- Percent surviving to times t=c(0,1,2)

```
summary(fit0, time = c(0, 1, 2))
```

```
## Call: survfit(formula = Surv(TTE, AE_any) ~ 1, data = dat_use)
##
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     0    180       0    1.000   0.000        1.000        1.000
##     1     67     113    0.372   0.036        0.308        0.450
##     2     59       8    0.328   0.035        0.266        0.404
```

Summary measures of S(t) in R: RMST

- Restricted mean survival time

```
print(fit0, rmean = 2)
```

```
## Call: survfit(formula = Surv(TTE, AE_any) ~ 1, data = dat_use)
##
##        n events rmean* se(rmean) median 0.95LCL 0.95UCL
## [1,] 180    132  0.836    0.0659   0.32   0.139   0.649
##     * restricted mean with upper limit =  2
```

```
print(fit0, rmean = 4)
```

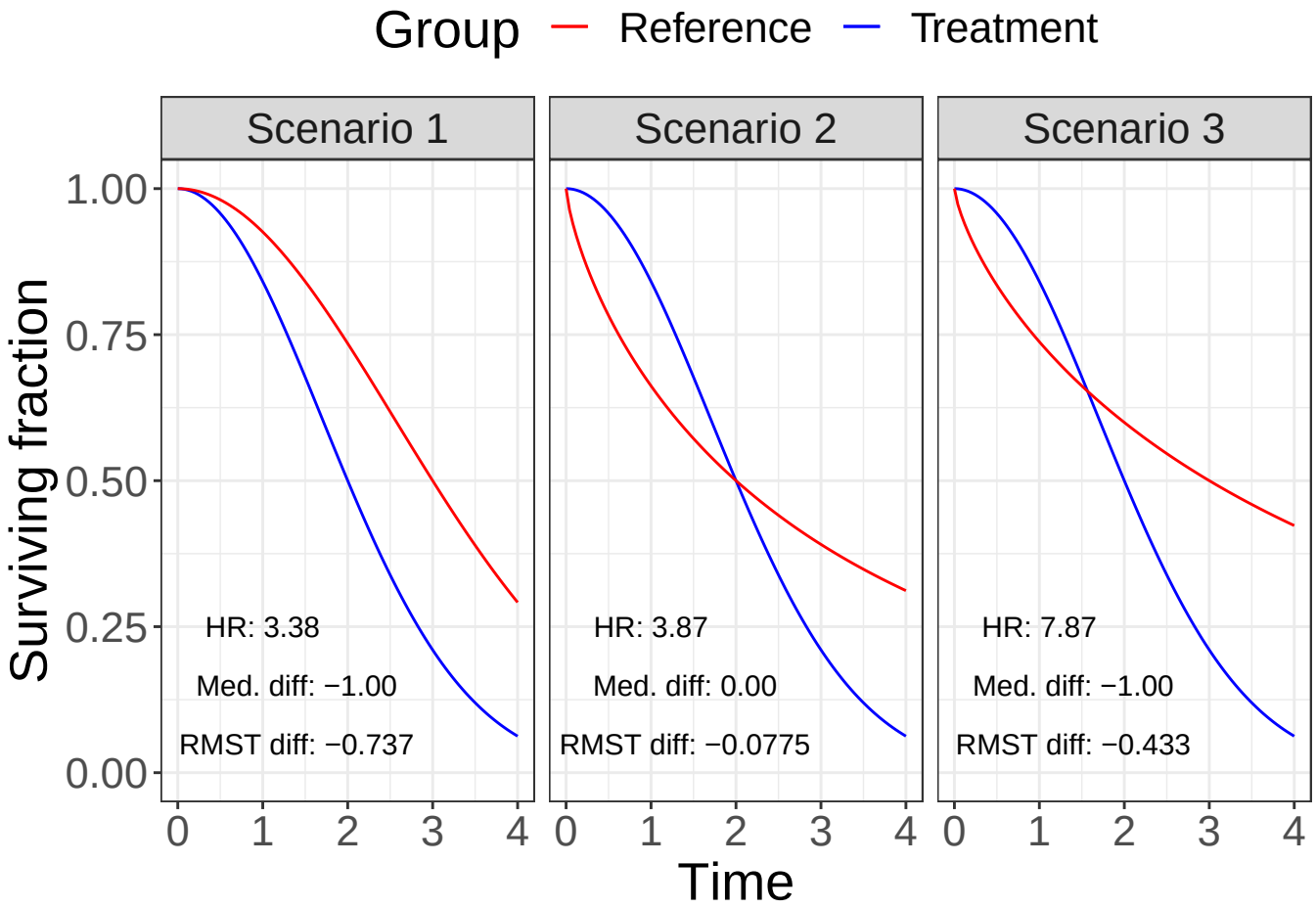
```
## Call: survfit(formula = Surv(TTE, AE_any) ~ 1, data = dat_use)
##
##        n events rmean* se(rmean) median 0.95LCL 0.95UCL
## [1,] 180    132    1.4     0.128   0.32   0.139   0.649
##     * restricted mean with upper limit =  4
```

Comparing two survival curves

- Hazard ratio
  - Most commonly used measure of effects
  - Closely tied to the Cox model
- Difference or ratio of median survival
  - Simple measure, easily understood
  - Connection to accelerated failure time models
- Difference or ratio of RMST
  - Re-emerging with treatments providing long-term cure fractions

Which one(s) you use depends on which aspects of the survival distribution are important

Are these survival distributions the same?



Testing for differences in survival functions

$H_0 : S_0(t) = S_1(t)$ vs $H_1 : S_0(t) \neq S_1(t)$

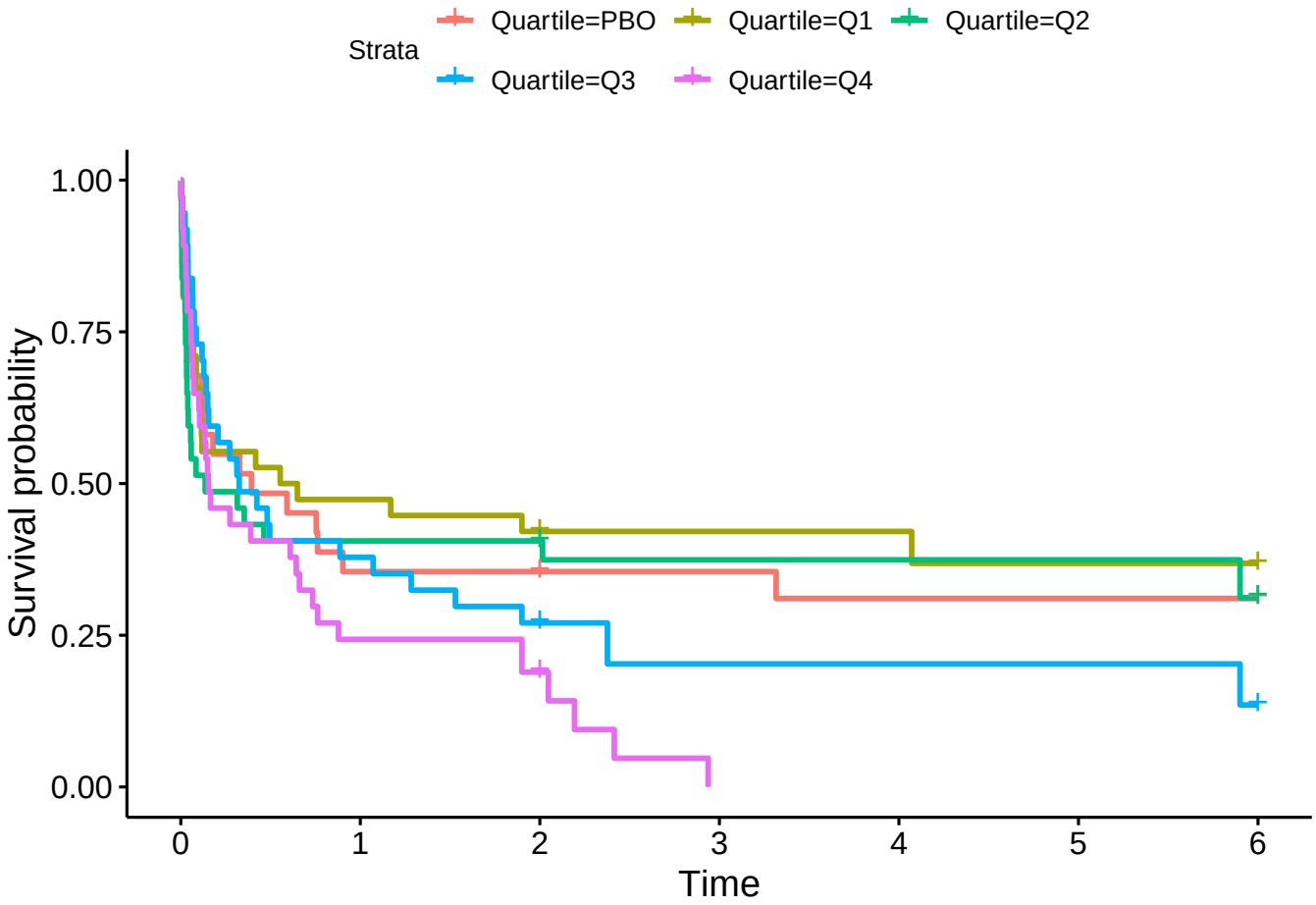
- Log rank test
  - Equivalent to Mantel-Haenszel test for binary data, where stratification is at unique event times
  - Most powerful test when the hazard ratio is constant (but applicable even if it is not)
  - Gives relatively higher weight to later differences in S(t)
- Generalized Wilcoxon test
  - A weighted version of the log-rank test with weights proportional to the number at risk
  - Gives relatively higher weight to early differences in S(t)
  - Thus, more sensitive when differences in survival occur early

Compare exposure quartiles using summary measures

```
km_exp <- survfit(Surv(TTE, AE_any) ~ Quartile, dat = dat_use)
print(km_exp)
```

```
## Call: survfit(formula = Surv(TTE, AE_any) ~ Quartile, data = dat_use)
##
##                 n events median 0.95LCL 0.95UCL
## Quartile=PBO 31     21  0.394  0.1172      NA
## Quartile=Q1  38     23  0.602  0.1061      NA
## Quartile=Q2  37     25  0.136  0.0392      NA
## Quartile=Q3  37     29  0.325  0.1503   1.529
## Quartile=Q4  37     34  0.154  0.1012   0.735
```

Compare exposure quartiles using summary measures



Compare exposure quartiles using log-rank test

```
survdiff(Surv(TTE, AE_any) ~ Quartile, data = dat_use)
```

```
## Call:
## survdiff(formula = Surv(TTE, AE_any) ~ Quartile, data = dat_use)
##
##                  N Observed Expected (O-E)^2/E (O-E)^2/V
## Quartile=PBO 31       21     23.9    0.3581     0.440
## Quartile=Q1  38       23     30.1    1.6677     2.171
## Quartile=Q2  37       25     26.5    0.0802     0.103
## Quartile=Q3  37       29     27.8    0.0555     0.071
## Quartile=Q4  37       34     23.8    4.3986     5.458
##
##  Chisq= 6.7  on 4 degrees of freedom, p= 0.2
```

Table of Contents

Semi-parametric models for TTE data

- Cox Proportional hazards model (the Cox model)
  - Baseline hazard is not estimated directly
- Connection to Poisson model
  - Breslow's formulation
- Piecewise exponential as a simplification
- Smooth baseline hazards
  - Provides a similar (though not identical) model with the ability to simulate

Cox PH model

Originally proposed by Cox in 1972 Cox (1972)

Model

$$h(t) = h_0(t) \exp(\theta_1 x_1 + \ldots \theta_p x_p)$$

Note: there is no intercept in the exponential ... why?

$h_0(t)$ is referred to as the baseline hazard

- $h_0(t) > 0$
- The baseline hazard function is not specified
  - We'll see that it does not need to be specified in order to estimate θ
- The covariate effects modify the hazard proportionately
  - The covariate model is linear in the parameters
  - Similar to the linear predictor we saw in logistic regression

Connection to survival function

Recall that

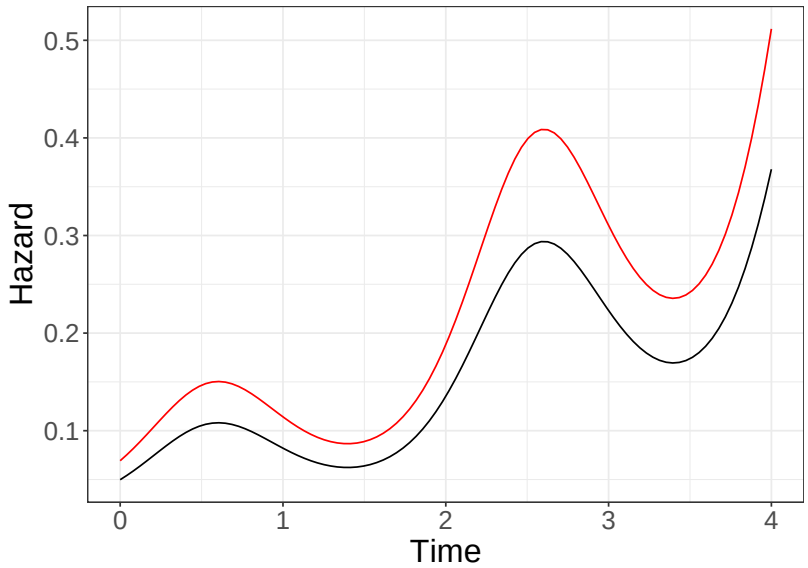
$$S(t) = \exp\left\{-\int_0^t h(s)\, ds\right\}$$

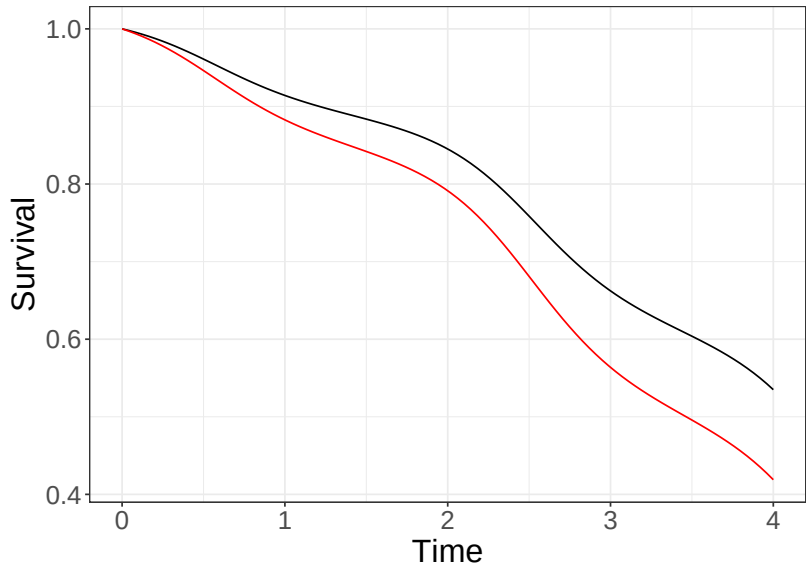
Then, under the Cox model we have

$$\begin{aligned} S(t) &= \exp\left\{-\int_0^t h_0(s)\exp(\theta_1 x_1)\, ds\right\} \\ &= \exp\left\{-\exp(\theta_1 x_1)\, H_0(t)\right\} \\ &= \{S_0(t)\}^{\exp(\theta_1 x_1)} \end{aligned}$$

where $S_0(t) = \exp\{-H_0(t)\}$ is the baseline survival function

Example of proportional hazards





Parameter estimation

Cox proposed using a partial likelihood approach, treating the baseline hazard as an infinite dimensional nuisance parameter.

This yields the likelihood function (assuming unique event times):

$$\ell(\theta|\mathbf{x}) = \prod_{i=1}^{k} \frac{\exp(\mathbf{x}_i^T\theta)}{\sum_{j\in R(i)} \exp(\mathbf{x}_j^T\theta)}$$

where

- there are k unique event times
- $R(j)$ is the set of subjects at risk for an event at $t(j)$

Note that the product is only over the k events, not over all subjects.

Parameter interpretation

Suppose we have the model

$$h(t) = h_0(t) \exp(\theta_1 x_1 + \theta_2 x_2)$$

where x_1 is binary and x_2 is continuous.

- Then, e^{θ_1} represents the hazard ratio comparing $x_1 = 1$ to $x_1 = 0$
  - θ_1 represents the log hazard ratio
- e^{θ_2} represents the hazard ratio for a one unit difference in x_2
  - $\exp(\theta_2 \times d)$ represents the hazard ratio for a d unit difference in x_2

Example model

As an example, let's fit the model:

$$h(t) = h_0(t) \exp\{\theta_1 \cdot Q_1 + \theta_2 \cdot Q_2 + \theta_3 \cdot Q_3 + \theta_4 \cdot Q_4\}$$

where the covariates $Q_1, \ldots, Q_4$ are indicators for exposure quartile

$$Q_j = \begin{cases} 1 & \text{CAVGSS in quartile } j \\ 0 & \text{otherwise} \end{cases}$$

Questions:

- What does $h_0(t)$ correspond to?
- What does θ_1 correspond to?

Estimation of the Cox model in R

To fit the Cox model in R, we'll use the `coxph` function from the `survival` package:

```
fit1 <- coxph(Surv(TTE, AE_any) ~ Quartile, data = dat_use)
```

- Similar LHS as used with `survfit` (for K-M estimate)
- RHS can include factor or continuous variables
  - Character variables are converted to factors

Output from `coxph`

```
print(fit1)
```

```
## Call:
## coxph(formula = Surv(TTE, AE_any) ~ Quartile, data = dat_use)
##
##                coef exp(coef) se(coef)      z      p
## QuartileQ1 -0.13566   0.87314  0.30201 -0.449 0.6533
## QuartileQ2  0.06989   1.07239  0.29661  0.236 0.8137
## QuartileQ3  0.18586   1.20425  0.28771  0.646 0.5183
## QuartileQ4  0.50397   1.65528  0.27983  1.801 0.0717
##
## Likelihood ratio test=6.39  on 4 df, p=0.1717
## n= 180, number of events= 132
```

- Hazard ratio comparing Q1 to placebo: 0.873 (0.483,1.58)

Estimation of the baseline cumulative hazard function from a Cox model

- Recall, estimation of covariate effects did not need an estimate of $h_0(t)$
- However, we can obtain a post-hoc estimate of $H_0(t)$ and $S_0(t)$

$$\widehat{H_0}(t) = \sum_{j:t(j)\leq t} \frac{D_j}{\sum_{i\in R(j)} \exp(\hat{\theta}_1 x_{1i} + \ldots \hat{\theta}_p x_{pi})}$$

where

- $t(j)$ are the ordered unique event times
- $R(j)$ is the set of subjects at risk for an event at $t(j)$
- D_j is the number of events at time $t(j)$

Baseline hazard estimation: intuition

$$\widehat{H_0}(t) = \sum_{j:t(j)\leq t} \frac{D_j}{\sum_{i\in R(j)} \exp(\hat{\theta}_1 x_{1i} + \ldots \hat{\theta}_p x_{pi})}$$

- With no covariates, this simplifies to the Nelson-Aalen estimator
- With covariates, the denominator is the counter-factual number of subjects at risk had all subjects been in the reference group

The baseline survival function is $\widehat{S_0}(t) = \exp\left\{-\widehat{H_0}(t)\right\}$

The subject-specific survival function is

$$\widehat{S_i}(t) = \left[\widehat{S_0}(t)\right]^{\exp(\hat{\theta}_1 x_{1i} + \cdots + \hat{\theta}_p x_{pi})}$$

Model evaluation for the Cox model

Our primary model evaluation tools will be:

- Comparing model predicted and observed survival
- Martingale and deviance residuals for functional form of covariate effects
- Assessing the PH assumption
- Concordance / Harrell's C-index

Comparing model predicted and observed survival

To extract predicted survival curves from a Cox model, we'll use the `survfit` function

```
# One line per unique covariate pattern
dat_pred <- dat_use %>%
  arrange(Quartile) %>%
  distinct(Quartile)

# Use survfit to extract predicted survival function
# survfit0 adds time=0 to the predictions; helps with plotting
cox_preds_fit1 <- survfit(fit1, newdata = dat_pred) %>%
  survfit0()
```

Comparing model predicted and observed survival

To extract predicted survival curves from a Cox model, we'll use the `survfit` function

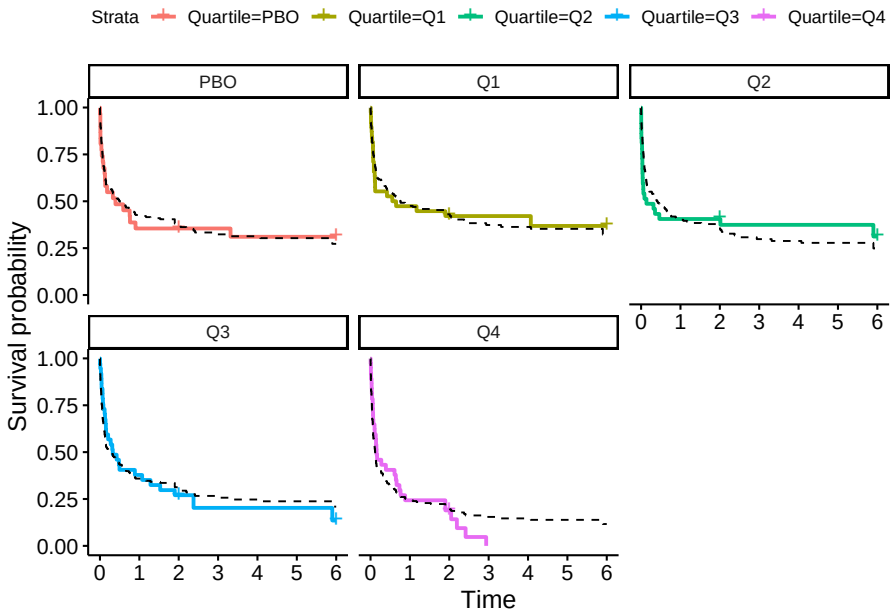
```
head(cbind(cox_preds_fit1$time, cox_preds_fit1$surv))
```

```
##                              1         2         3         4         5
## [1,] 0.0000000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2,] 0.0008052808 0.9952417 0.9958441 0.9948982 0.9942727 0.9921360
## [3,] 0.0019346783 0.9904835 0.9916857 0.9897981 0.9885509 0.9842966
## [4,] 0.0022448385 0.9857236 0.9875233 0.9846981 0.9828328 0.9764792
## [5,] 0.0023020837 0.9809590 0.9833542 0.9795947 0.9771146 0.9686788
## [6,] 0.0030770114 0.9761791 0.9791693 0.9744769 0.9713839 0.9608783
```

Comparing model predicted and observed survival

```
survival_preds_fit1 <- cox_preds_fit1$surv %>%
  as.data.frame() %>%
  mutate(time = cox_preds_fit1$time) %>%
  pivot_longer(cols = -time) %>%
  mutate(
    group = as.numeric(name),
    Quartile = paste0(dat_pred$Quartile[group])
  )

ggsurvplot(survfit(Surv(TTE, AE_any) ~ Quartile, data =
  data = dat_use
)$plot +
  geom_step(
    data = survival_preds_fit1,
    aes(x = time, y = value), linetype = "dashed"
  ) +
  facet_wrap(~Quartile)
```



Martingale residuals

The Martingale residual is defined as

$$M_i = \delta_i - \widehat{H}_i(T_i^*)$$

where

$$\delta_i = \begin{cases} 1 & \text{event} \\ 0 & \text{censored} \end{cases}$$

and T^* is the observed event or censoring time

Note

Because $H(t)$ is is expected number of events according to the model, the Martingale residual is an ”observed - expected” type of residual.

Often asymmetric due to the fact that $M_i \in (-\infty, 1)$

Deviance residuals

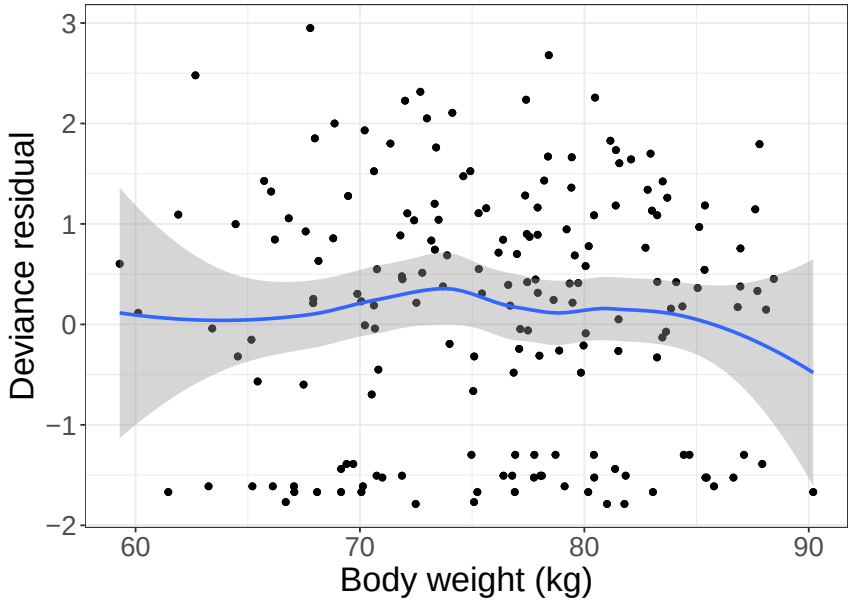
The deviance residual is defined as

$$D_i = \text{sign}(M_i)\sqrt{-2\left(M_i + \delta_i \log \widehat{H}_i(T_i^*)\right)}$$

- More symmetric than Martingale residuals
- Roughly have mean=0 and sd=1

Uses for Martingale and Deviance residuals

- Plot vs linear predictor to assess exponential link
- Plot vs covariate to assess functional form
  - If model is correct, should see no trends




```
dat_use <- dat_use %>%
  ungroup() %>%
  mutate(dev_resids = residuals(fit1, type = "deviance"))

ggplot(dat_use, aes(x = BWT, y = dev_resids)) +
  geom_point() +
  geom_point() +
  geom_smooth() +
  labs(x = "Body weight (kg)", y = "Deviance residual")
```

Assessing PH assumption

- Plot standardized Schoenfeld residuals vs time
  - Shows how coefficient changes with time
  - Flat line if PH model is correct
- Test for significance of including time-by-covariate interactions in the model
- Compare predicted and observed survival

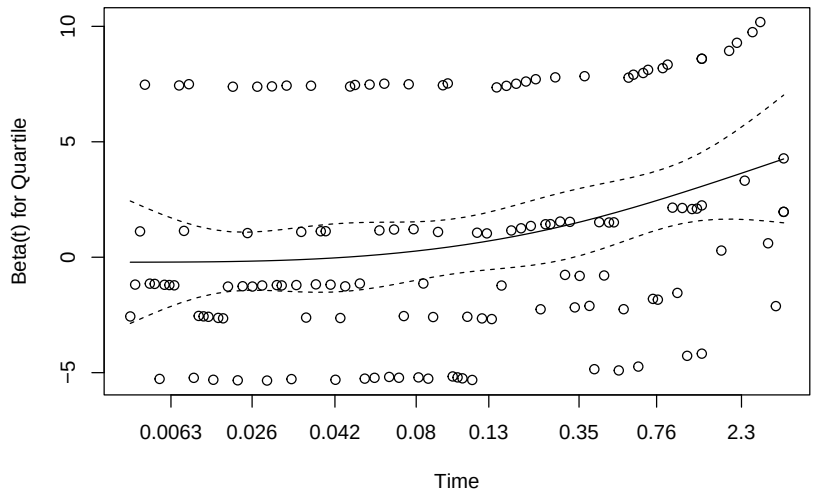
Assessing PH assumption: example

```
assess_ph_fit1 <- cox.zph(fit1)

print(assess_ph_fit1)
```

```
##            chisq df       p
## Quartile    23.3  4 0.00011
## GLOBAL      23.3  4 0.00011
```

```
plot(assess_ph_fit1)
```



- This is the score test for adding a time-by-covariate interaction.
  - Indicates a violation of the PH assumption
  - Also suggested by the plot of the standardized Schoenfeld residuals vs time

Concordance as a measure of model fit

Concordance is a measure of how well the predicted risk of an event aligns with the observed risk.

Imagine you have two randomly selected subjects *i* and *j* with

- Covariates x_i and x_j
- Risk scores: $\eta_i = x_i^T \hat{\theta}$ and $\eta_j = x_j^T \hat{\theta}$
- (True) event times: T_i and T_j

Then concordance (c-index) estimates: $P(T_i > T_j \mid \eta_i < \eta_j)$

Harrell et al. Harrell et al. (1982) proposed a method for handling censored data.

A model with higher c-index should provide better predictions.

Cox model as a Poisson regression

It turns out that we can get the same likelihood if we view the data as arising from a Poisson process with intervals defined by the unique event times. Whitehead (1980)

- Divide time into non-overlapping intervals with segments defined by observed event times
- Constant baseline hazard within each interval

Note

This yields another way of thinking about the Cox model as a data generating model.

- Data are arising from a continuous baseline hazard which we estimate with piecewise constants
- Piecewise constant hazard = piecewise exponential model

Piecewise exponential as an approximation to the Cox model

- This means that one approximation to a Cox model is
  - Piecewise exponential with predefined intervals
  - The more intervals, the closer the result is to Cox model

The model is given by

$$h(t) = h_0(t) \exp(\theta_1 x_1 + \cdots + \theta_p x_p)$$

where

$$h_0(t) = \lambda_j \text{ for } t \in [\tau_{j-1}, \tau_j)$$

e.g., Ibrahim, Chen, and Sinha (n.d.)

Piecewise exponential as an approximation to the Cox model

The model is given by

$$h(t) = h_0(t) \exp(\theta_1 x_1 + \cdots + \theta_p x_p)$$

where

$$h_0(t) = \lambda_j \text{ for } t \in [\tau_{j-1}, \tau_j)$$

- How to define the time intervals depends on
  - Clinical knowledge (e.g., transplant French, Thomas, and Wang (2012))
  - Time-varying predictors
  - Expected number and timing of events
- This formulation provides a semi-parametric model from which we can simulate survival data

Smooth non-parametric baseline hazard functions

- An alternative semi-parametric model is to estimate the baseline hazard using smoothing splines Royston and Parmar (2002)
- Can be fitted using the `flexsurv` R package and in `brms`
- Royston and Parmar Royston and Parmar (2002) propose $g(S(t)) = s(t) + \beta^T x$
  - $s(t)$ is a smooth function modeled using cubic smoothing splines
  - $g(S) = \log(-\log(S))$ corresponds to a proportional hazards model
  - $g(S) = \log(1/S - 1)$ corresponds to a proportional odds model
- `brms` implementation uses M-splines for the baseline hazard
  - M-splines are non-negative and integrate to I-splines
  - Simplifies calculations

Extensions to standard Cox model

- Stratified Cox model
  - Allows different baseline hazard by strata
  - $h(t) = h_{0,s}(t) \exp(\theta_1 x_1 + \cdots + \theta_p x_p)$
  - Often used in primary analysis of TTE endpoints in clinical trials (e.g., stratify by region)
- Time-varying covariates
  - Allows for covariates to be constant over intervals defined by the data
  - Doesn't allow for continuously varying covariates
  - Frequently used when conditioning on an intermediate event (e.g., Stanford heart transplant data)

Introduction to Parametric models

- The Cox model was designed to estimate effects of covariates on the time-to-event distribution.
  - With that focus, the baseline hazard isn't of much interest.
- However, in pharmacometric modeling we are often interested in modeling the full time-to-event distribution
  - Both the baseline hazard and covariate effects
- This will be our focus for the next section
  - Accelerated failure time models in R and Stan
  - General hazard models in Stan

Traditional parametric models

Many of the commonly used models fall under the accelerated failure time framework (Wei, Wei (1992))

$$\log(T) = \mu + \theta \boldsymbol{X} + \epsilon$$

Distribution for ϵ	***Distribution for T***
Extreme value	Weibull
Extreme value (scale=1)	Exponential
Normal	Log-normal
Logistic	Log-logistic

But any distribution for T that has support on values of $T \geq 0$ is suitable, including Gompertz, Gamma, Generalized Gamma, etc.

Nice, conceptual introductions to (parametric) TTE modeling are by Holford (2013) and Bradburn et al. (2003)

AFT vs PH models

- Covariate effects in AFT models are fundamentally different than in PH models
- In a PH model, a covariate scales the hazard function
  - $h(t) = h_0(t)\exp(\theta x)$
  - $S(t) = S_0(t)^{\exp(\theta x)}$
- In an AFT model, a covariate scales time
  - $\log T = \mu + \theta x + \epsilon$
  - $S(t) = S_0(ct)$, where $c = \exp(\theta x)$ is the **acceleration factor**
  - The covariate effect is on the percentiles of the distribution
  - $\frac{\text{Percentile when } x=1}{\text{Percentile when } x=0} = \frac{1}{c}$ for all percentiles

Features of common distributions

- Exponential
  - Hazard is constant
  - One parameter
- Weibull, Gompertz, and Gamma
  - Two parameters (scale/location and shape)
  - Hazard in monotonically increasing or decreasing
- Log-logistic and log-normal
  - Two parameters (scale/location and shape)
  - Hazard is uni-modal
  - Falling, or arc (rising then falling)
- Generalized Gamma
  - Three parameters (scale/location and shape)
  - Monotonic (increasing or decreasing); arc; bathtub

Likelihood for parametric survival models

Assuming that censoring times are independent of event times, then the individual contribution to the likelihood function is

$$L_i(\theta) = \begin{cases} f(T_i^*) & \text{for } \delta_i = 1 \\ S(T_i^*) & \text{for } \delta_i = 0 \end{cases}$$

and the likelihood function is

$$\begin{aligned} L(\theta) &= \prod_{i=1}^{n} f(T_i^*)^{\delta_i} S(T_i^*)^{1-\delta_i} \\ &= \prod_{i=1}^{n} h(T_i^*)^{\delta_i} S(T_i^*) \end{aligned}$$

Fitting parametric TTE models in R

- The `survreg()` function in the `survival` package
- The `flexsurvreg()` function in the `flexsurv` package

We generally use the `flexsurv` package because there are more distributions available

Model evaluation

- Residual plots
  - Similar use as with the Cox model
- Simulation-based diagnostics
  - VPCs for survival and hazard functions
  - NPDEs

Numeric model comparison

- AIC and friends
- C-index

Table of Contents

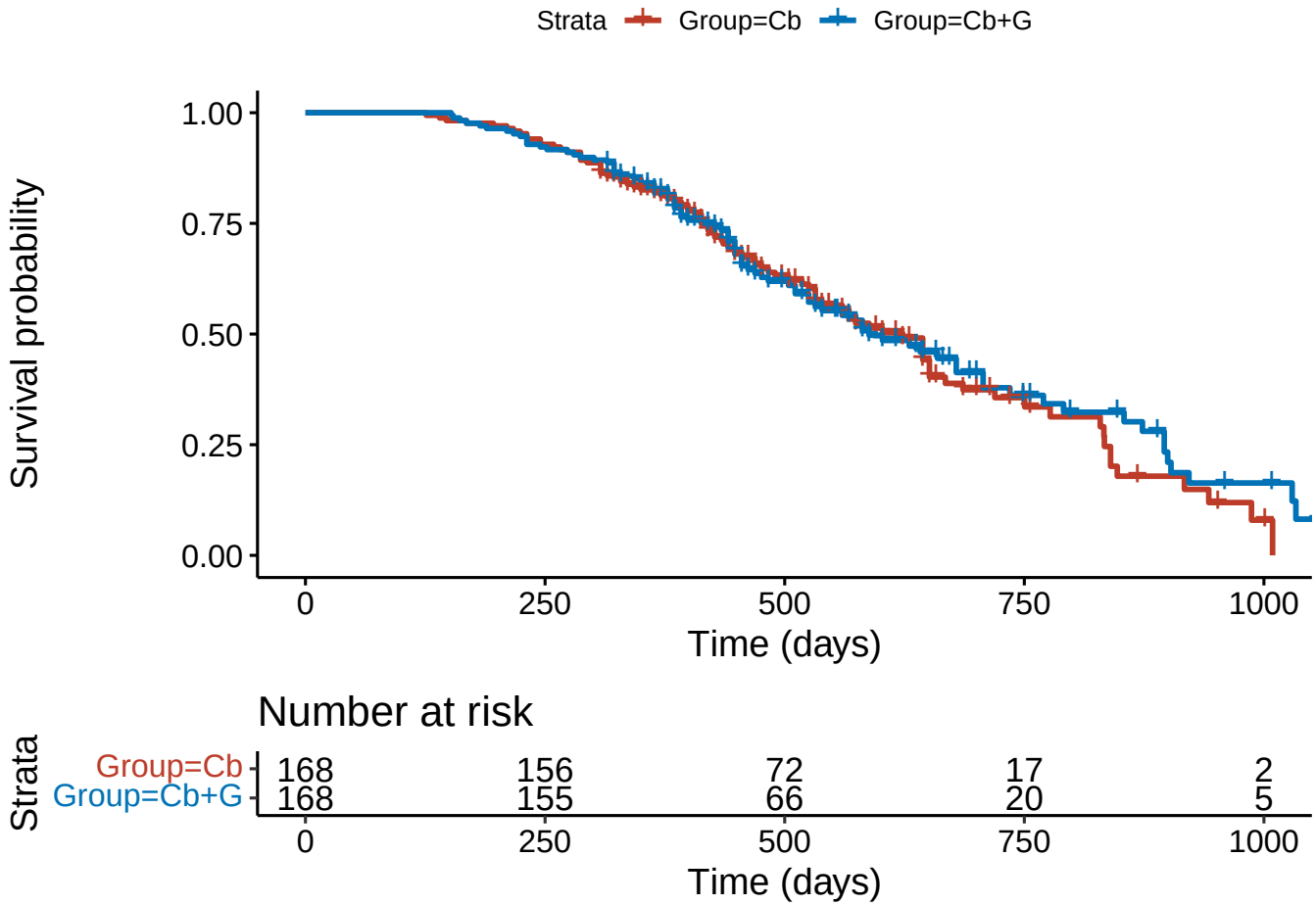
A little notation

- There are two time-to-event processes happening:
  - T = time to event of interest
  - C = time to censoring
- With right censoring, we observe
  - $T^* = \min(T, C)$
  - $\delta = I(T \leq C)$
- We are trying to estimate the distribution of T, but we observe T^*
  - We'll return to this when discussing model diagnostics
- Typical to assume that T and C are independent

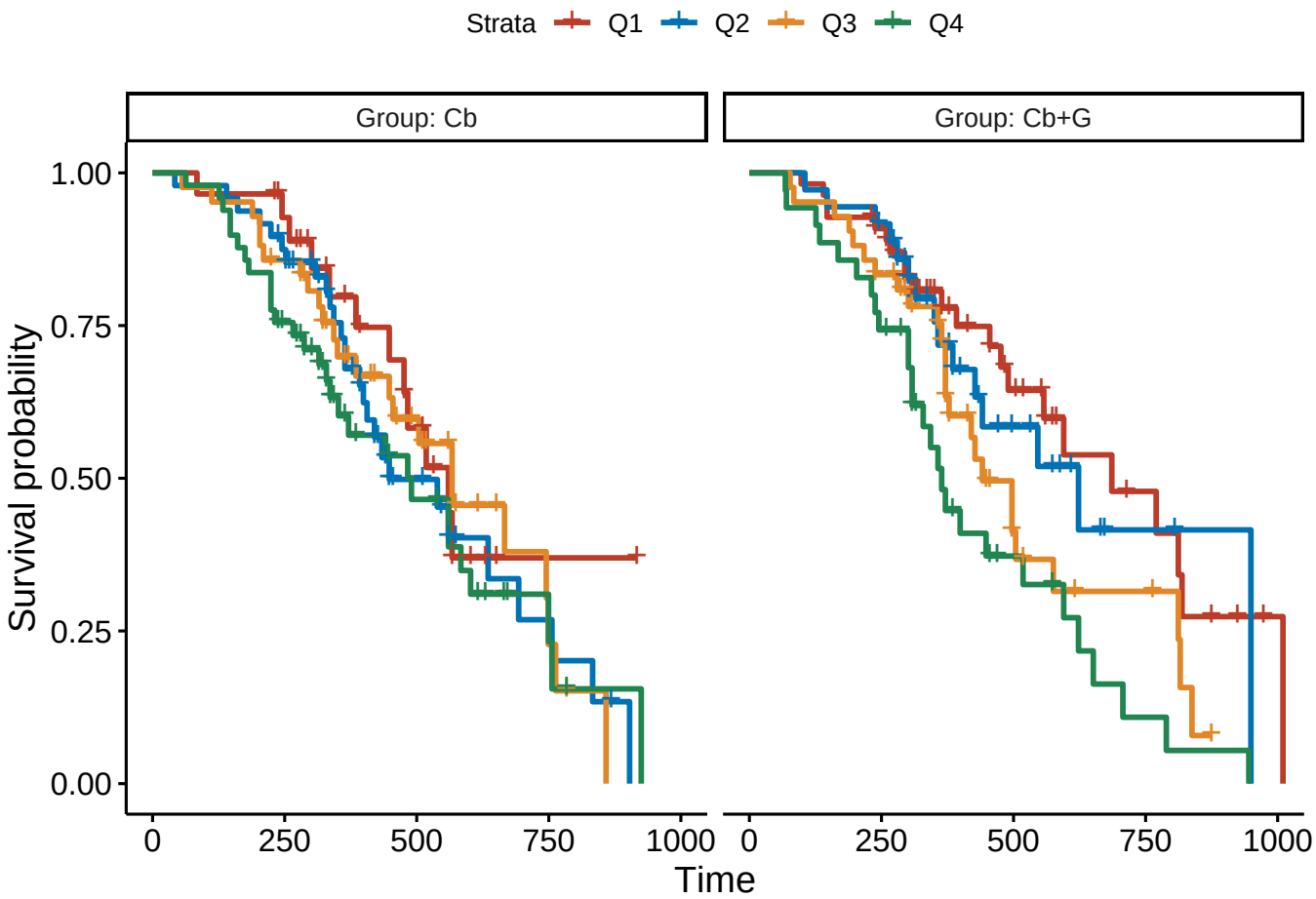
New dataset

- **Models for change in tumour size, appearance of new lesions and survival probability in patients with advanced epithelial ovarian cancer** (Zecchin et al. 2016)
  - DDMORE repository submission IDs: DDMODEL00000217, DDMODEL00000218
  - Data simulated from these models
- Original study
  - Patients with platinum-sensitive recurrent ovarian cancer
  - Randomly assigned to receive gemcitabine plus carboplatin (Cb+G) or carboplatin alone (Cb), every 21 days
  - Primary objective was to compare progression-free survival (PFS)
- We will analyze overall survival (OS) and the relationship between tumor changes and OS, using simulated data

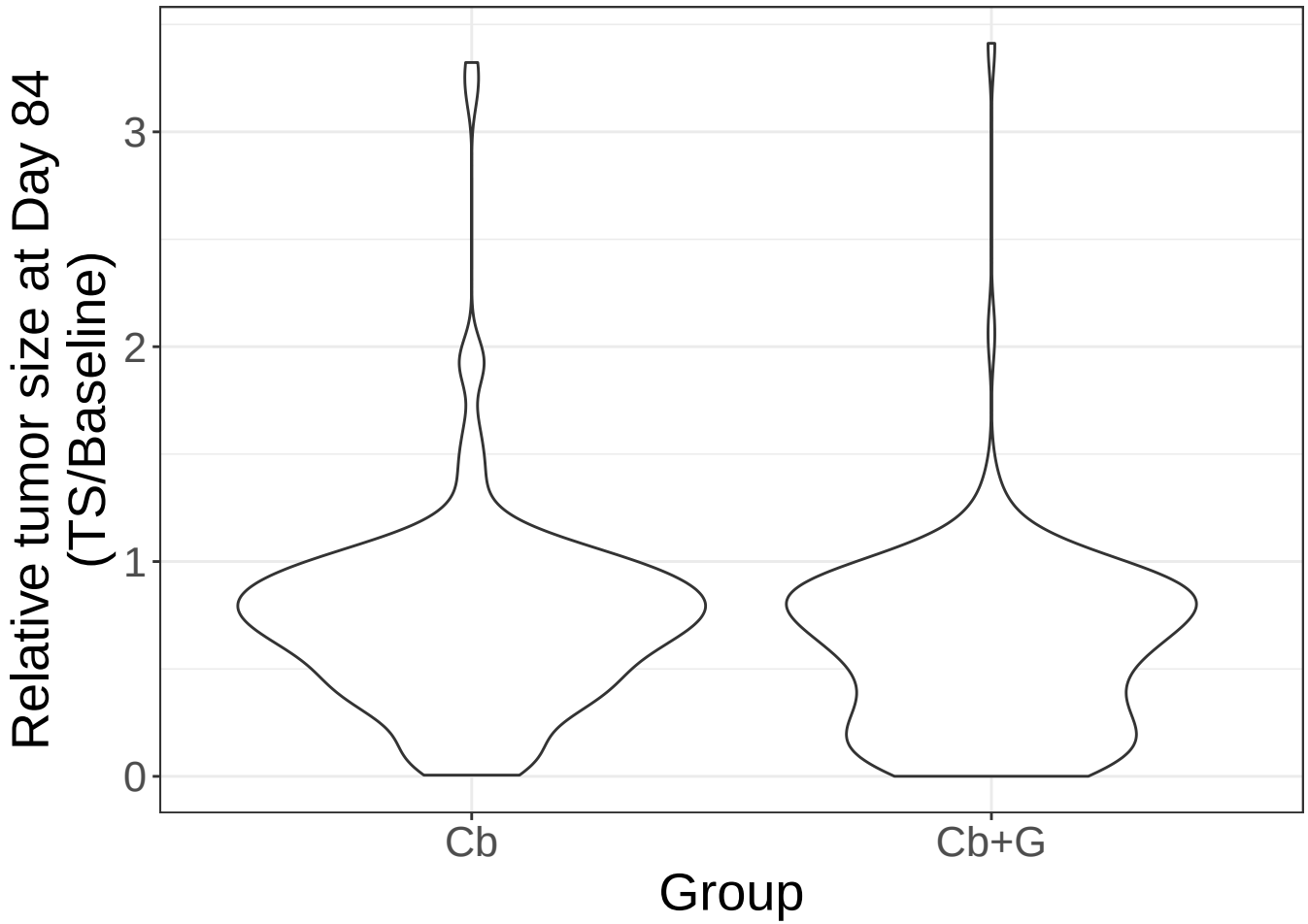
OS by treatment group



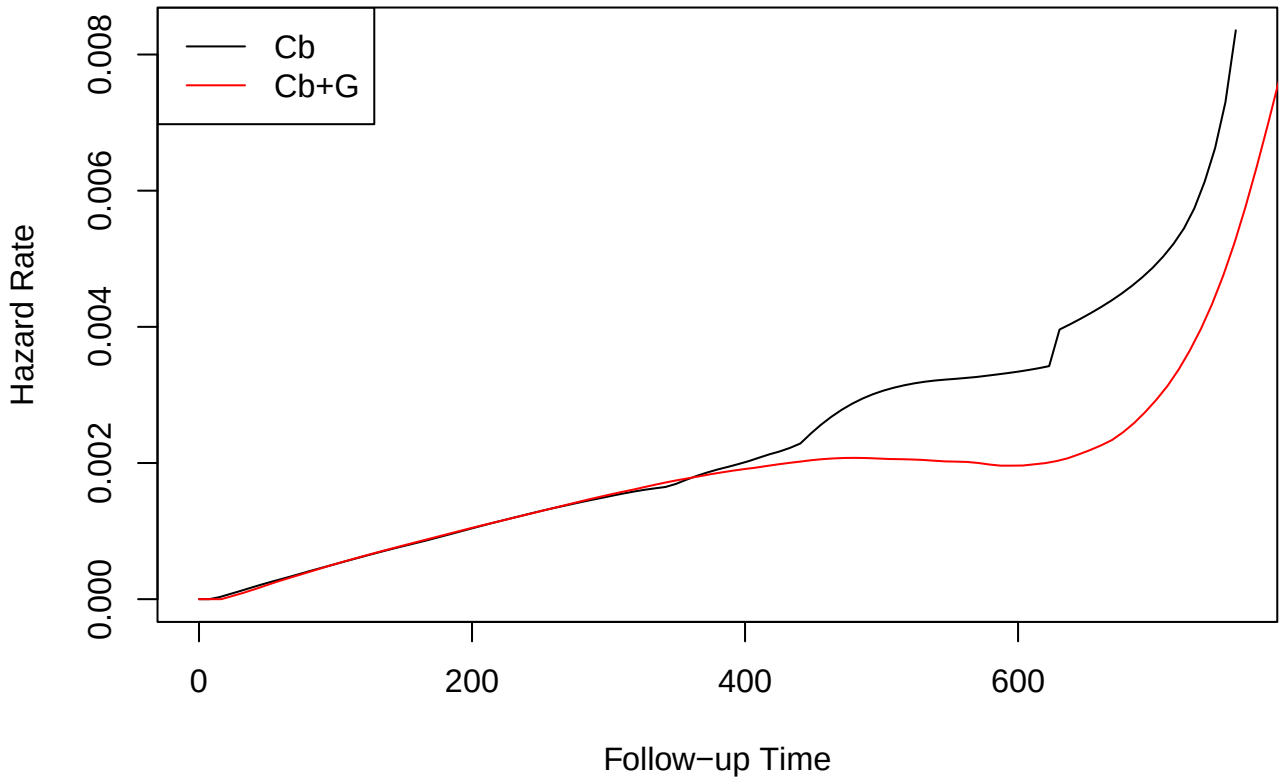
Landmarked OS at Day 84 by change in tumor size and group



Distribution of change in tumor size to Day 84



What hazard function might make sense?



Let's start by fitting a Weibull model as a function of relative tumor size (RTS)

```
weibull_prior <- c(
  brms::prior(lognormal(0, 3), class = "shape"),
  brms::prior(normal(0, 3), class = "b")
)

fit_weibull <- brms::brm(TIME | cens(1 - DV) ~ I(rts84 - 1),
  data = dos84,
  prior = weibull_prior,
  family = brms::weibull()
)
```

The model is

$$\log \text{TIME}_i = \theta_0 + \theta_1 \times (RTS_i - 1) + \epsilon_i$$

where $\epsilon \sim$ extreme value distribution

θ_0 corresponds to the mean OS on the log scale when RTS=1 ($\exp(\theta_0)$ is the median OS); θ_1 is the acceleration factor

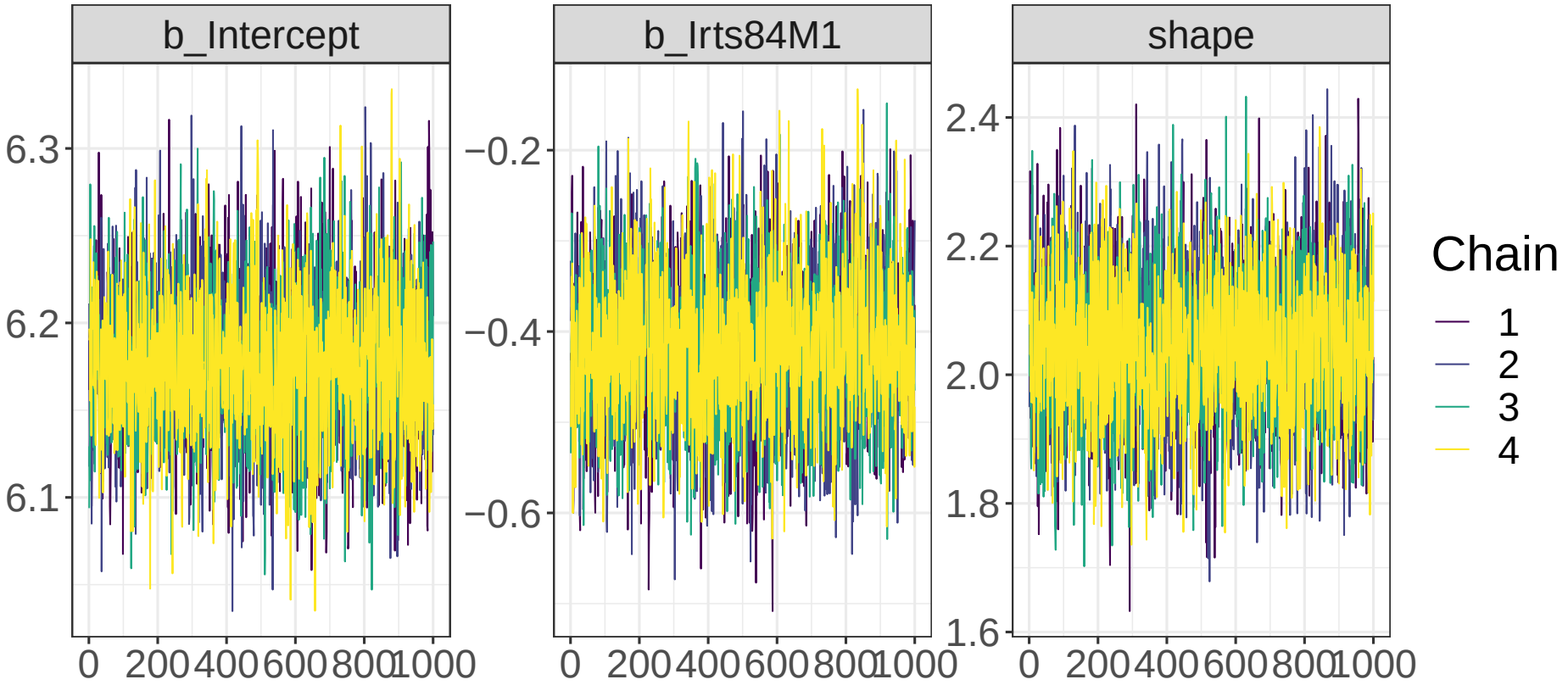
Output from Weibull model

```
##  Family: weibull
##   Links: mu = log; shape = identity
## Formula: TIME | cens(1 - DV) ~ I(rts84 - 1)
##    Data: dos84 (Number of observations: 336)
##   Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##          total post-warmup draws = 4000
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept     6.18      0.04     6.10     6.26 1.00     4331     2988
## Irts84M1     -0.42      0.08    -0.58    -0.25 1.00     2765     3096
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## shape     2.04      0.11     1.82     2.27 1.00     3595     3271
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

Convergence assessments

- Rhat values all look good (previous slide)
- Trace plots look good

```
brms::mcmc_plot(fit_weibull, type = "trace")
```



Model evaluation

- Residual plots
  - Similar use as with the Cox model
- Simulation-based diagnostics
  - Visual predictive checks (VPCs) for survival and hazard functions
  - NPDEs

Posterior predictive checks

- Simulate many replicates of the DV using the estimated model and observed predictors
  - Accounting for censoring process
- Determine summary statistic(s) of interest
  - K-M estimate of S(t)
  - Non-parametric estimate of h(t)
  - Mean covariate value among subjects at risk
- Calculate summary statistic for observed data
- Calculate summary statistic for each simulated replicate
- Plot distribution(s) of summary statistics
- Overlay observed value

Simulate survival times from model

```
weibull_sims <- add_predicted_draws(
  newdata = dos84 %>% select(ID, rts84, rts84_f, ECOG),
  fit_weibull,
  value = "survival_time"
)
```

```
## # A tibble: 6 x 7
## # Groups:   ID, rts84, rts84_f, ECOG, .row [1]
##      ID rts84 rts84_f  ECOG  .row .draw survival_time
##   <dbl> <dbl> <chr>   <dbl> <int> <int>         <dbl>
## 1     1 0.629 Q2          1     1     1          680.
## 2     1 0.629 Q2          1     1     2          400.
## 3     1 0.629 Q2          1     1     3          577.
## 4     1 0.629 Q2          1     1     4          581.
## 5     1 0.629 Q2          1     1     5          413.
## 6     1 0.629 Q2          1     1     6         1019.
```

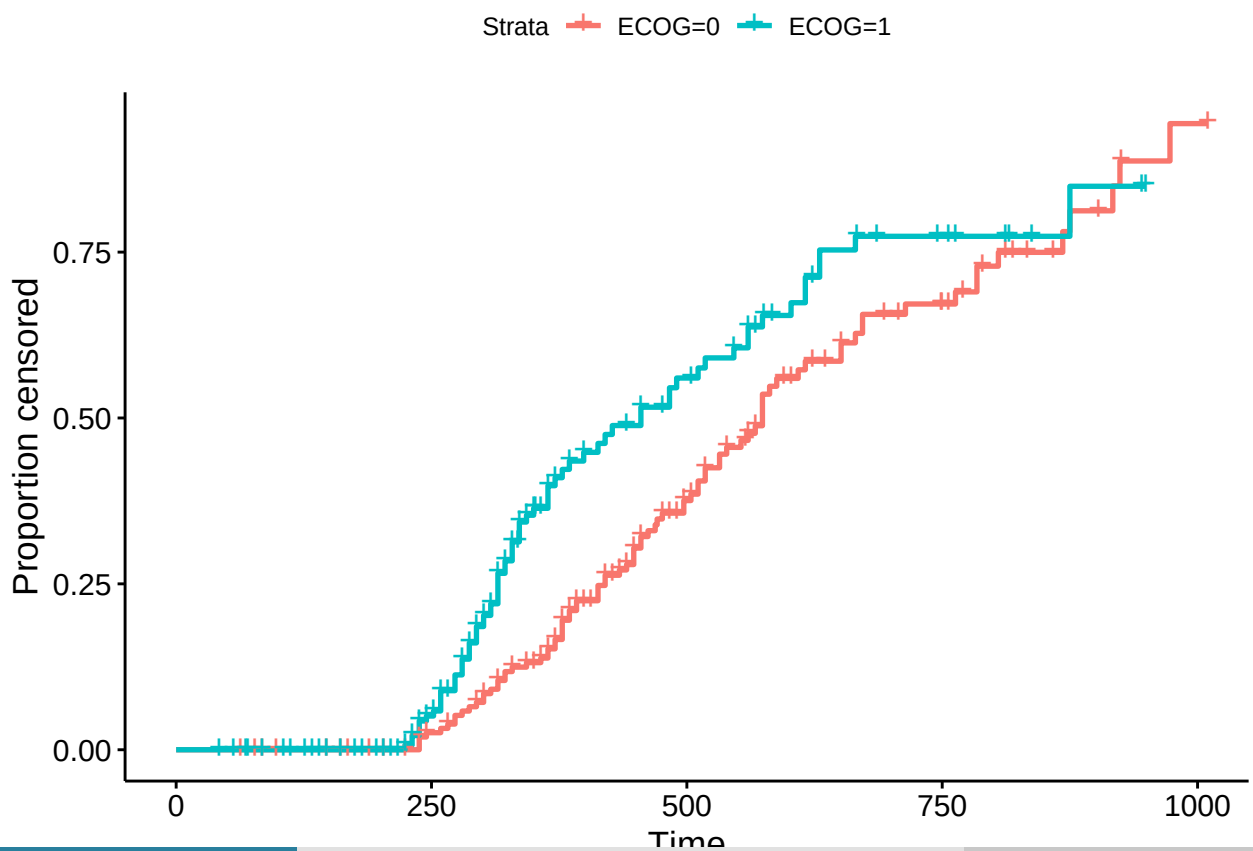
These are simulations of T. To reflect the changing risk-set it is often advisable to also simulate censoring times to get to $T^* = \min(T, C)$

Options for distribution of C

- Kaplan-Meier estimator
- Cox model
- Parametric model
- Do not use observed event times to censor simulated times
  - Mixture of event and censoring distributions

Time to censoring of OS

```
ggsurvplot(survfit(Surv(TIME, 1 - DV) ~ ECOG, data = dos84),
  fun = "event",
  ylab = "Proportion censored",
  data = dos84
)
```



Fit log-normal model for censoring distribution

```
fit_censoring <- brms::brm(
  TIME | cens(DV) ~ ECOG,
  data = dos84,
  family = brms::lognormal()
)
```

```
##  Family: lognormal
##   Links: mu = identity; sigma = identity
## Formula: TIME | cens(DV) ~ ECOG
##    Data: dos84 (Number of observations: 336)
##   Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##          total post-warmup draws = 4000
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept     6.36      0.04     6.28     6.44 1.00     3153     2829
## ECOG         -0.21      0.06    -0.33    -0.10 1.00     3524     3016
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma     0.45      0.03     0.40     0.50 1.00     2653     2791
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

Simulate censoring times and derive the event time

```
censoring_sims <- add_predicted_draws(
  newdata = dos84 %>% select(ID, rts84, rts84_f, ECOG),
  fit_censoring,
  value = "censoring_time"
)

event_sims <- weibull_sims %>%
  left_join(censoring_sims) %>%
  mutate(
    event_time = pmin(survival_time, censoring_time),
    delta = survival_time < censoring_time
  )
```

```
## # A tibble: 6 x 10
## # Groups:   ID, rts84, rts84_f, ECOG, .row [1]
##      ID rts84 rts84_f  ECOG  .row .draw survival_time censoring_time event_time
##   <dbl> <dbl> <chr>   <dbl> <int> <int>         <dbl>          <dbl>      <dbl>
## 1     1 0.629 Q2          1     1     1          680.           322.       322.
## 2     1 0.629 Q2          1     1     2          400.           492.       400.
## 3     1 0.629 Q2          1     1     3          577.           629.       577.
## 4     1 0.629 Q2          1     1     4          581.           283.       283.
## 5     1 0.629 Q2          1     1     5          413.          1026.       413.
## 6     1 0.629 Q2          1     1     6         1019.           510.       510.
## # i 1 more variable: delta <lgl>
```

Summary statistic: Kaplan-Meier estimate of S(t) stratified by RTS quartile

```
vpc_stat_km <- function(.data, pred_times = NULL) {
  fit <- survfit(Surv(time, event) ~ rts84_f, data = .data)
  if (is.null(pred_times)) {
    pred_times <- c(0, sort(unique(fit$time)))
  }
  preds <- summary(fit, times = pred_times)

  data.frame(pred_times = preds$time, preds = preds$surv,
             group = preds$strata)
}
```

Calculate K-M estimator for observed and simulated data

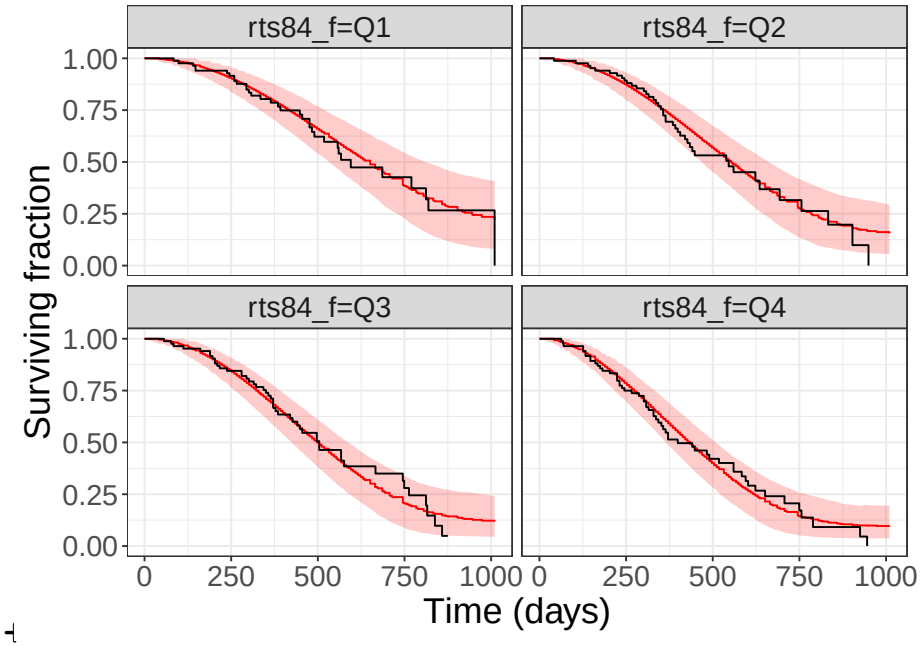
```
obs_surv <- vpc_stat_km(dos84 %>% mutate(time = TIME, event = DV))
```

Apply the summary statistic to each simulated dataset

```
sim_surv <- event_sims %>%
  mutate(time = event_time, event = as.numeric(delta)) %>%
  nest(data = -.draw) %>%
  mutate(km_est = map(data,
    ~ vpc_stat_km(., pred_times = sort(unique(obs_surv$pred_times))))) %>%
  select(-data) %>%
  unnest(cols = km_est)
```

Plot survival function VPC

```
p <- sim_surv %>%
  group_by(pred_times, group) %>%
  summarise(
    med = mean(preds),
    lcl = quantile(preds,
                   probs = 0.05),
    ucl = quantile(preds,
                   probs = 0.95)
  ) %>%
  ggplot(aes(x = pred_times)) +
  geom_step(aes(y = med), color = "red") +
  geom_ribbon(aes(ymin = lcl, ymax = ucl), fill = "red", alpha = 0.2) +
  geom_step(data = obs_surv, aes(y = preds)) +
  facet_wrap(~group) +
  ylim(0, 1) +
  labs(x = "Time (days)", y = "Surviving fraction")
```



Summary statistic: hazard function

```
vpc_stat_hazard <- function(.data, .maxtime = NULL) {
  grid <- seq(0, .maxtime, length = 101)

  if (!is.null(.maxtime)) {
    fit <- with(.data, muhaz(time, event,
                             min.time = 0, max.time = .maxtime))
  } else {
    fit <- with(.data, muhaz(time, event,
                              min.time = 0))
  }

  # Impute at grid times in case muhaz uses different estimation points
  # -- Impute NA if .maxtime is beyond last event time
  haz <- approx(x = fit$est.grid, y = fit$haz.est, xout = grid, rule = 1)

  data.frame(pred_times = grid, preds = haz$y)
}
```

Apply to observed and simulated data

We will estimate the hazard until only 5% of subjects remain at risk.

```
endtime <- dos84 %>%
  arrange(TIME) %>%
  slice(floor(.95 * n())) %>%
  pull(TIME)
```

```
obs_hazard <- dos84 %>%
  rename(time = TIME, event = DV) %>%
  nest(data = -rts84_f) %>%
  mutate(hazard = map(data,
    ~ vpc_stat_hazard(.data = ., .maxtime = endtime))) %>%
  select(-data) %>%
  unnest(hazard)
```

Apply to simulated data

Apply the summary statistic to each simulated dataset

```
sim_hazard <- event_sims %>%
  mutate(time = event_time, event = as.numeric(delta)) %>%
  filter(.draw <= 500) %>%
  arrange(.draw, rts84_f) %>%
  nest(data = -c(.draw, rts84_f)) %>%
  mutate(hazard = map(data, ~ vpc_stat_hazard(., .maxtime = endtime))) %>%
  select(-data) %>%
  unnest(cols = hazard)
```

Plot hazard function VPC

```
p <- sim_hazard %>%
  group_by(pred_times, rts84_f) %>%
  summarise(
    med = mean(preds, na.rm = TRUE),
    lcl = quantile(preds,
                   probs = 0.05, na.rm = T
    ucl = quantile(preds,
                    probs = 0.95, na.rm =
  ) %>%
  ggplot(aes(x = pred_times)) +
  geom_step(aes(y = med), color = "red") +
  geom_ribbon(aes(ymin = lcl, ymax = ucl), fill = "red", alpha = 0.2) +
  geom_line(data = obs_hazard, aes(y = preds)) +
  facet_wrap(~rts84_f) +
  labs(x = "Time (days)", y = "Hazard")
```

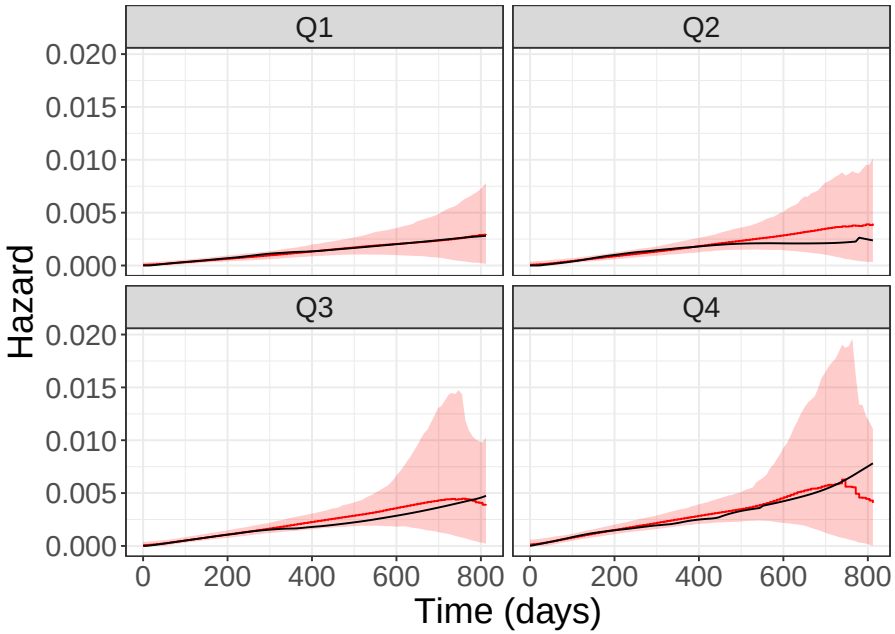


Table of Contents

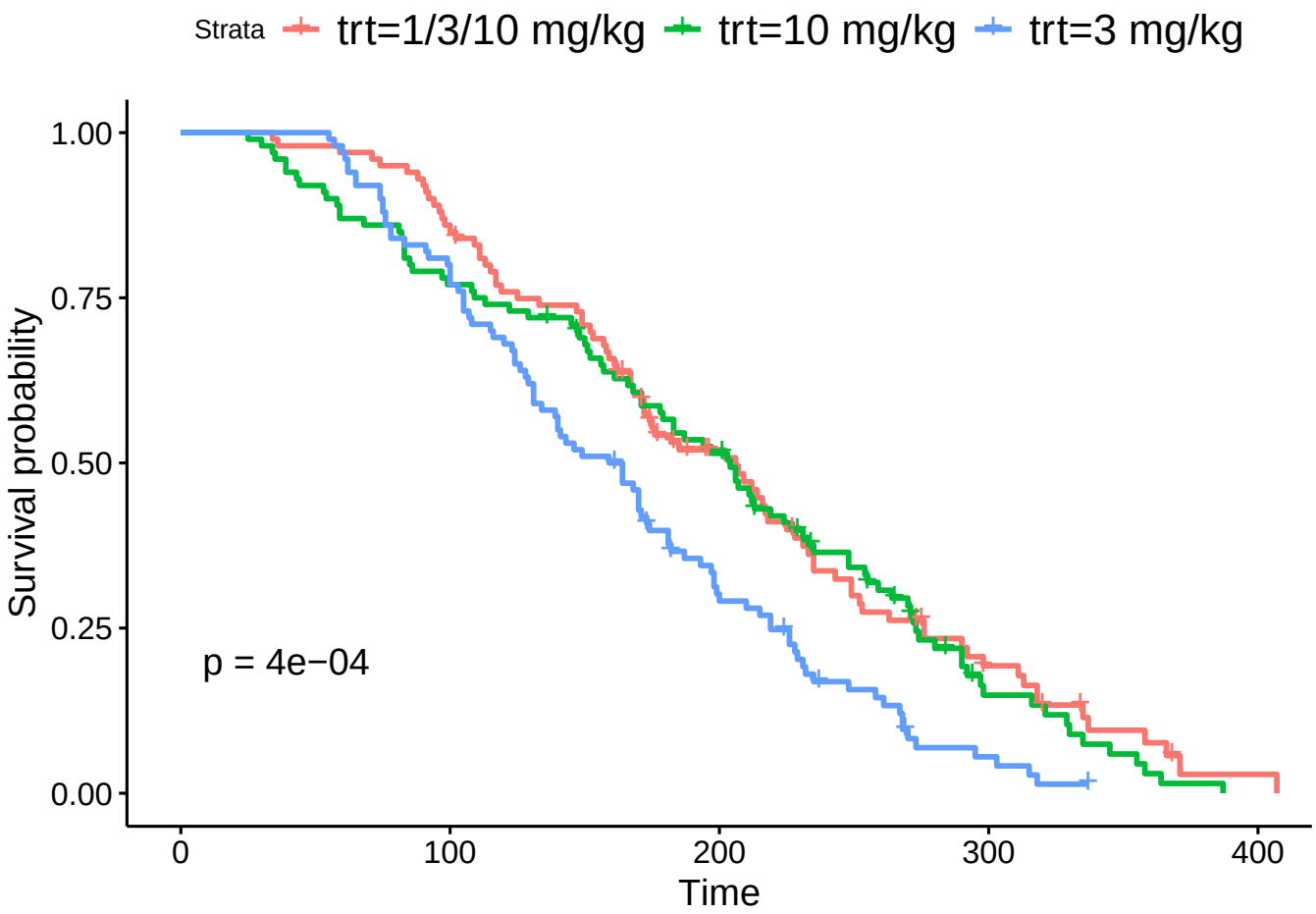
Example: progression-free survival in oncology trial

- Clinical trial with 3 treatment arms, with 3-week cycles (Q3W)
  1. Titration:
     - 1 mg/kg for 1 cycle
     - 3 mg/kg for 1 cycle
     - 10 mg/kg for remaining cycles
  2. 3 mg/kg Q3W
  3. 10 mg/kg Q3W
- 100 subjects per arm
- Endpoint is progression-free survival
  - Time from start of treatment to disease progression or death

Dataset structure: include time to event/censoring with per-cycle Cavg

```
## # A tibble: 2,722 x 8
##       id cycle  week  dose  cavg quartile   tte event
##    <dbl> <dbl> <dbl> <dbl> <dbl> <chr>    <dbl> <dbl>
##  1     1     1     0     1  1.30 Q4         298     1
##  2     1     2     3     3  3.89 Q4         298     1
##  3     1     3     6    10 13.0  Q4         298     1
##  4     1     4     9    10 13.0  Q4         298     1
##  5     1     5    12    10 13.0  Q4         298     1
##  6     1     6    15    10 13.0  Q4         298     1
##  7     1     7    18    10 13.0  Q4         298     1
##  8     1     8    21    10 13.0  Q4         298     1
##  9     1     9    24    10 13.0  Q4         298     1
## 10     1    10    27    10 13.0  Q4         298     1
## # i 2,712 more rows
```

KM by treatment: Some separation between groups

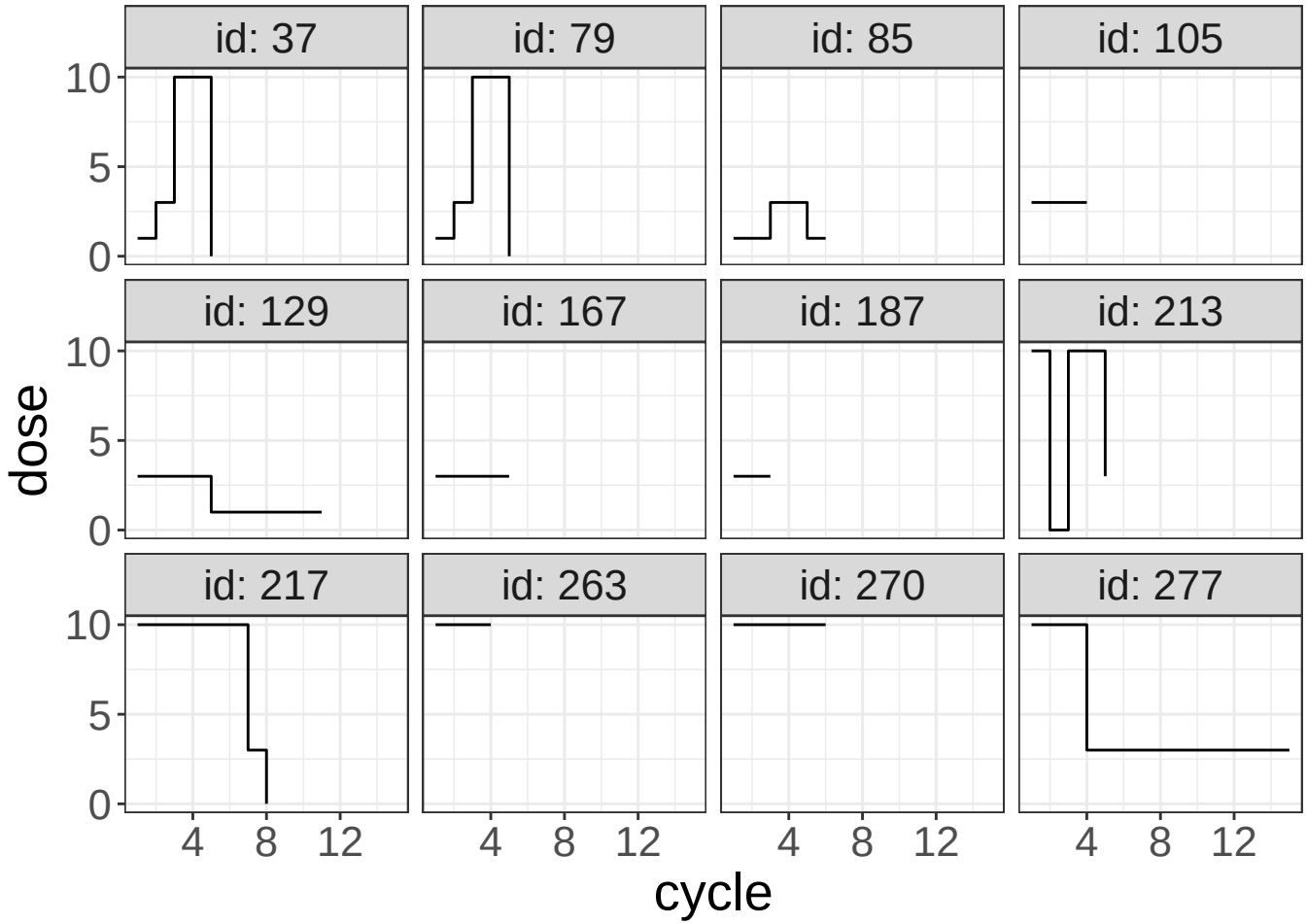


What to use as an exposure metric?

- “This is a key component of any analysis and may include dose, concentration, time-averaged concentration, time above a threshold, or area-related metrics”
- **“Be careful of dose adjustments and dropouts, and their effect on exposure metrics”**
- “In choosing a metric consider whether the relationship may be a direct effect (like nausea/vomiting) or a time delay effect like tumor growth”

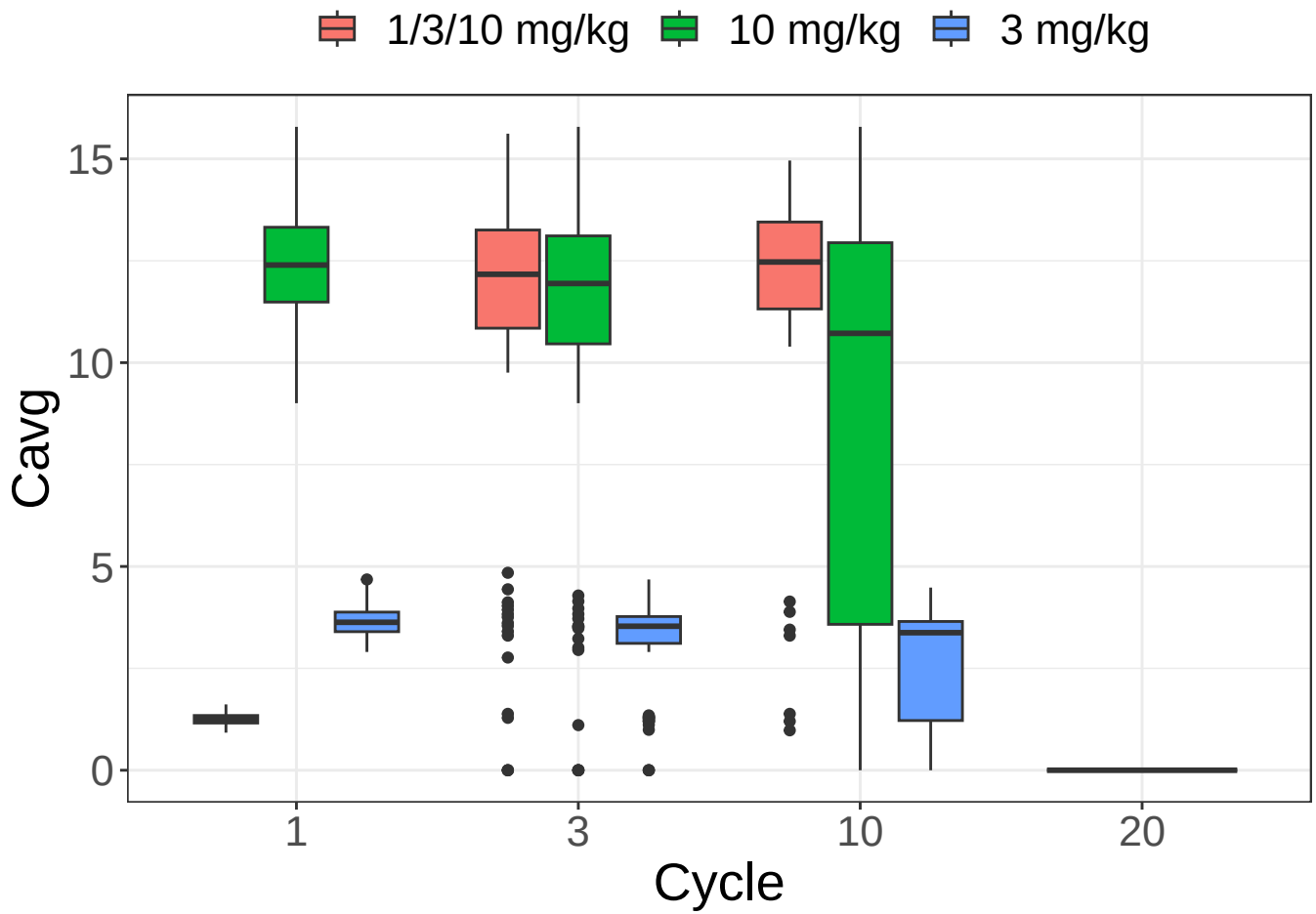
(Ruiz-Garcia et al. 2023)

This study has a lot of dose adjustments and dropouts

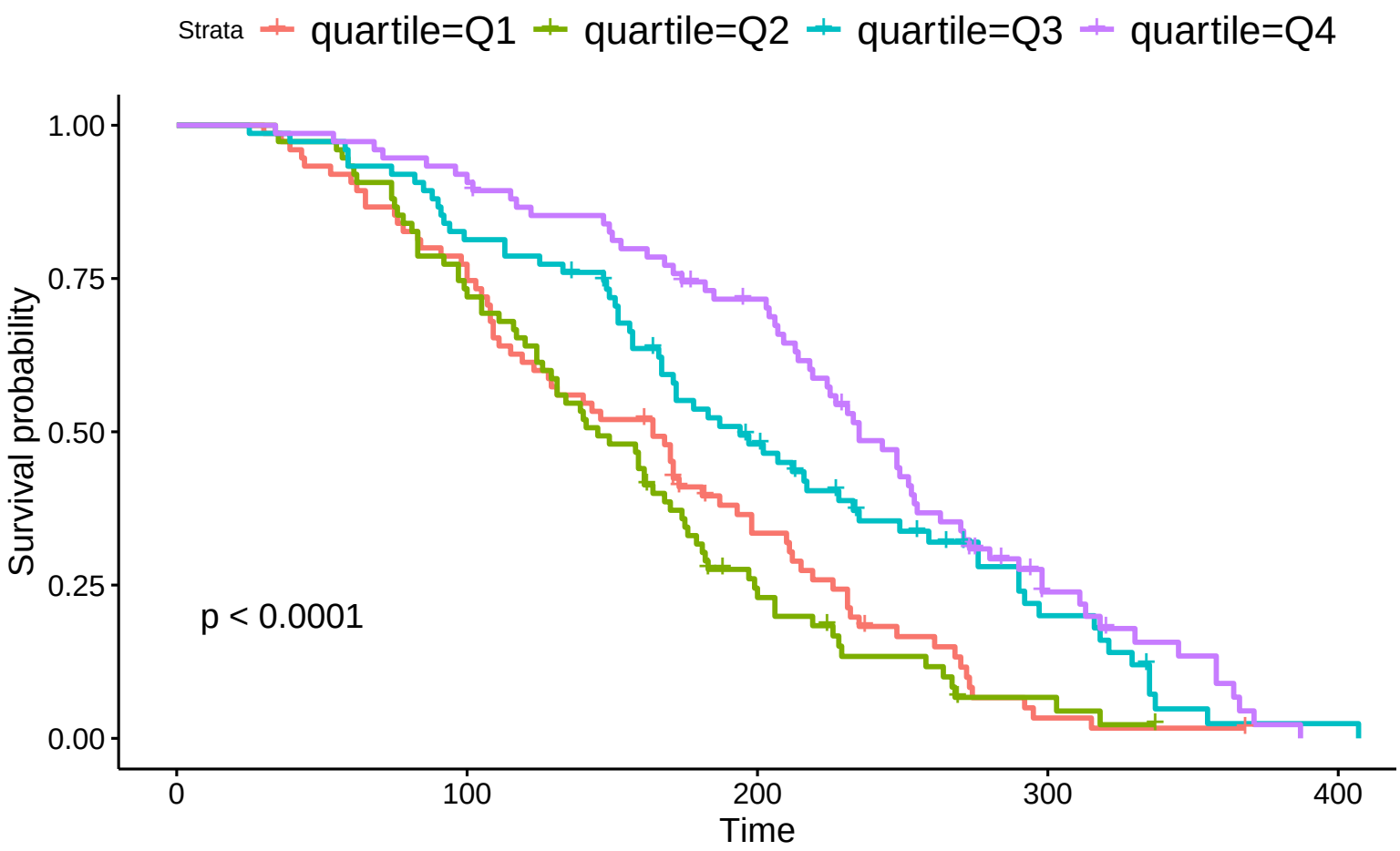


Distributions of Cavg by cycle

A single exposure metric won't cut it, but we need something to stratify KM plot. Let's go with Cycle 3.



KM plot by Cycle 3 Cavg shows clear exposure-response



For time-varying exposures we return to the piecewise exponential (PE) model

Recall: the PE model is given by

$$h(t) = h_0(t) \exp(\theta_1 x_1 + \cdots + \theta_p x_p)$$

where

$$h_0(t) = \lambda_j \text{ for } t \in [\tau_{j-1}, \tau_j)$$

- How to define the time intervals depends on
  - Clinical knowledge (e.g., transplant French, Thomas, and Wang (2012))
  - Time-varying predictors
  - Expected number and timing of events
- This formulation provides a semi-parametric model from which we can simulate survival data

Typically we explore a variety of linear and nonlinear functional relationships

$$h(t_{ij}) = h_0(t_{ij}) \exp(f(C_{ij}, \theta) + X_i^T \gamma_1)$$

where:

- $h(t_{ij})$ is the hazard at time t_{ij} for patient i.
- $f(C_{ij})$ is the functional relationship between exposure and the hazard, relative to the baseline
- θ is a vector of parameters for the exposure sub-model
- C_{ij} is exposure metric of patient i at time t_{ij}
- X_i is the vector of baseline covariates of patient i
- γ_1 are the main effects (on the log scale) of X_i on the hazard

Examples of functional forms for ER

$$f(C_i) = C_i\alpha_1 + C_i X_i^T \gamma_2 \quad \text{linear}$$

$$f(C_i) = \log(C_i)\alpha_1 + \log(C_i) X_i^T \gamma_2 \quad \text{log-linear}$$

$$f(C_i) = (\text{Emax} + X_i^T \gamma_2)\frac{C_i}{\text{EC50} + C_i} \quad \text{Emax}$$

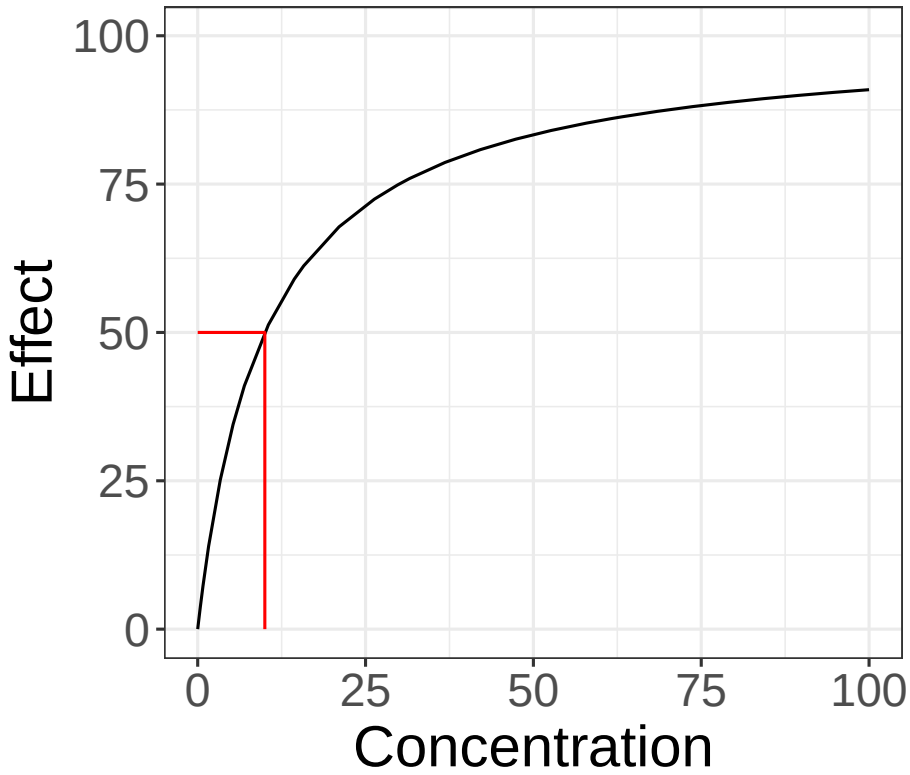
$$f(C_i) = (\text{Emax} + X_i^T \gamma_2)\frac{C_i^h}{\text{EC50}^h + C_i^h} \quad \text{Sigmoidal Emax}$$

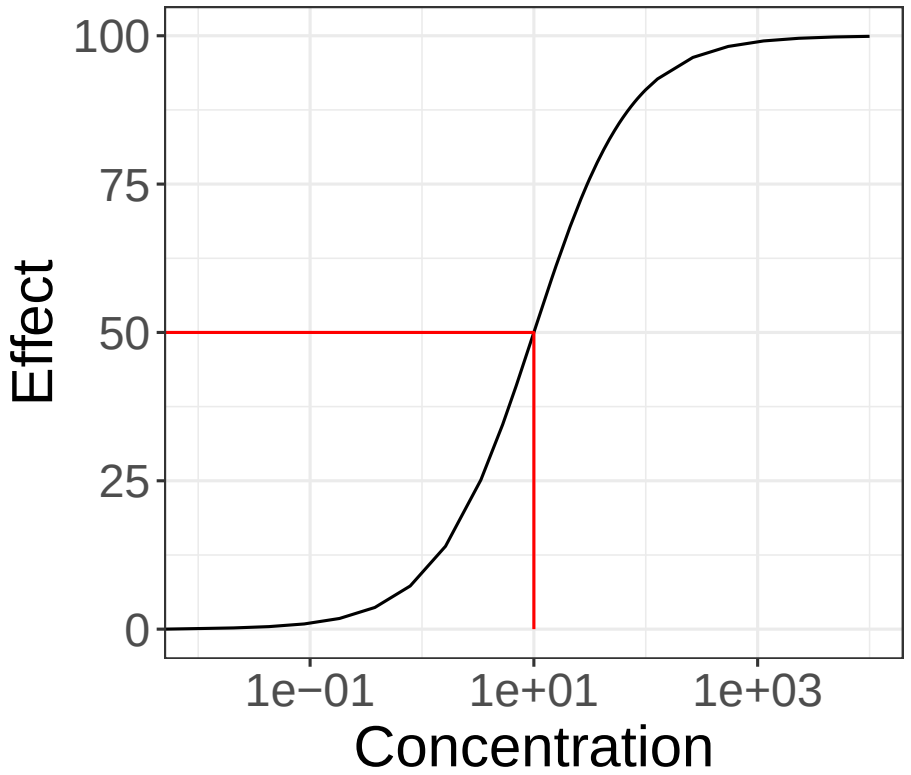
where:

- γ_2 is the vector of coefficients corresponding to the interaction effect of each covariate with exposure

The Emax model arises from binding of a drug to a receptor

$$f(C_i) = (\text{Emax} + X_i^T \gamma_2) \frac{C_i}{\text{EC50} + C_i}$$





Selecting priors for baseline hazards in PE model

$$\lambda_k \sim \text{Gamma}\left(\hat{\lambda}_k/c, 1/c\right)$$

where:

- $\hat{\lambda}_k$ is the prior mean
  - more on this soon
- c quantifies dispersion
  - a large value (c = 100) ensures a non-informative prior
- $c\hat{\lambda}_k$ is the variance

(Qing, Thall, and Yuan 2023)

Selecting priors for baseline hazards in PE model

To select prior means $\hat{\lambda}_k$:

1. Approximate the PE model with a Weibull distribution
   - obtain estimates $\hat{\alpha}$ and $\hat{\beta}$
2. Solve for prior means using the sub-interval average:

$$\hat{\lambda}_k = \frac{1}{\tau_k - \tau_{k-1}} \int_{\tau_{k-1}}^{\tau_k} \hat{\lambda}(t)\mathrm{d}t = \frac{\tau_k^{\hat{\alpha}} - \tau_{k-1}^{\hat{\alpha}}}{\hat{\beta}^{\hat{\alpha}}(\tau_k - \tau_{k-1})}$$

(Qing, Thall, and Yuan 2023)

Priors for ER parameters are weakly-informative Normals

$$\text{Emax} \sim N(0, 0.5)$$
$$\text{EC50} \sim N(\hat{\mu}_{\text{C1}}, 2\hat{\sigma}_{\text{C1}})$$

where $\hat{\mu}_{\text{C1}}$ and $\hat{\sigma}_{\text{C1}}$ are the mean and SD of Cycle 1 exposures, respectively

Stan model: data (1)

```
// intervals contructed using the following distinct times
// event/censoring times, hazard intervals times,
// and times of when time-varying covariate (exposure)

// number of intervals
int<lower=0> n_intervals;
// id defined such that EXPOSURE[id], int_length[id] correspond
// to exposures and interval length values of subject id
array[n_intervals] int id;
// length of each interval
// lag(cumsum(int_length[id])) denotes start time of each interval
// (starting at 0)
// cumsum(int_length[id]) denotes end time of each interval of subject id
vector[n_intervals] int_length;
```

Stan model: data (2)

```
// censoring times and events, N is number of subjects
int<lower=0> N;
// censoring indicator
// 1 no censoring...event
// 0 right censoring
// 2 interval censoring
array[N] int censoring;
// index where left and right censoring occurred
array[N] int rcensindex;
array[N] int lcensindex;
array[N] int begin;
array[N] int end;
array[N] int numrows;
```

Stan model: data (3)

```
// ID of hazard which corresponds to each interval
// lambda[id_hazard] should corresponds to hazard in intervals
array[n_intervals] int id_hazard;

// exposure metric
// EXPOSURE[id] corresponds to the exposure values at distinct times
// of subjectid
vector[n_intervals] EXPOSURE;
```

Stan model: data (4)

```
// Hyperparameters of EC50 metric
real EC50m;
real EC50s;

// Hyperparameter for Emax
real Emax_mean;
real<lower=0> Emax_sd;

// Hyperparameter for betaexp
real betaexp_mean;
real<lower=0> betaexp_sd;

// number of hazard intervals
int<lower=0> J;
// hyperparameters for hazard,.i.e lambda[j]
vector[J] lambda_hat;
// dispersion parameter
real<lower=0> c;
```

Stan model: parameters

```
array[J] real<lower=0> lambda;
real Emax;
real<lower=0> EC50;
```

Stan model: priors

```
// Priors on ER parameters
target += normal_lpdf(Emax | Emax_mean, Emax_sd);
target += normal_lpdf(EC50 | EC50m, EC50s);

// Prior on lambda
// doi:10.1002/pst.2256
for (j in 1:J) target += gamma_lpdf(lambda[j] |  lambda_hat[j] / c, 1/c);
```

Stan model: log probability density

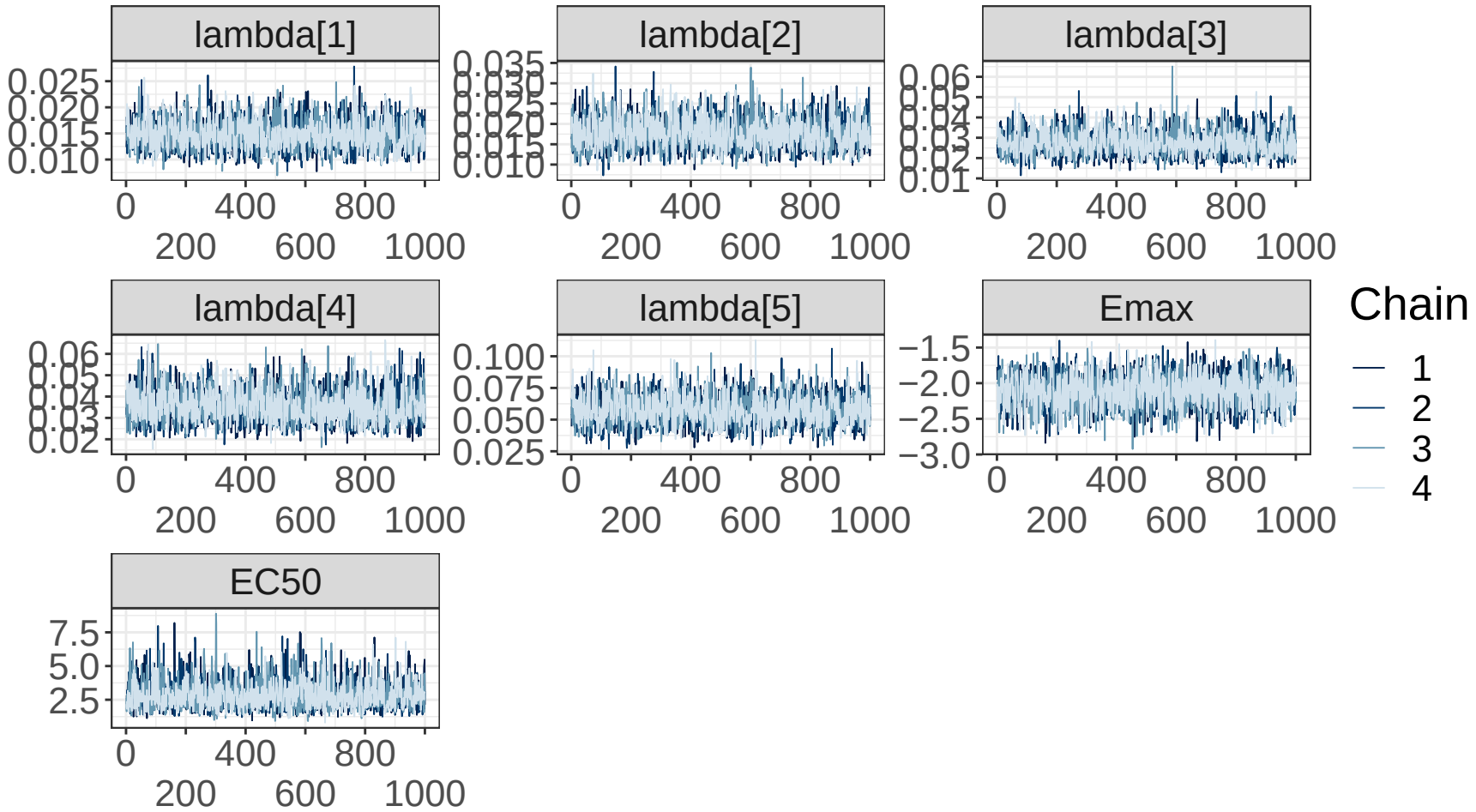
```
for (i in 1:N) {
  vector[numrows[i]] lp =
    Emax * EXPOSURE[begin[i]:end[i]] ./ (EC50 + EXPOSURE[begin[i]:end[i]]);
  vector[numrows[i]] lambda_vec =
    (to_vector(lambda)[id_hazard])[begin[i]:end[i]];
  vector[numrows[i]] llcont =
    -exp(lp) .* lambda_vec .* int_length[begin[i]:end[i]];
  if (censoring[i] == 1) {
      target += sum(head(llcont, rcensindex[i]-1));
      target += lp[rcensindex[i]] + log(lambda_vec[rcensindex[i]]);
  }
  if (censoring[i] == 0) {
      target += sum(head(llcont, rcensindex[i]-1));
  }
}
```

“”

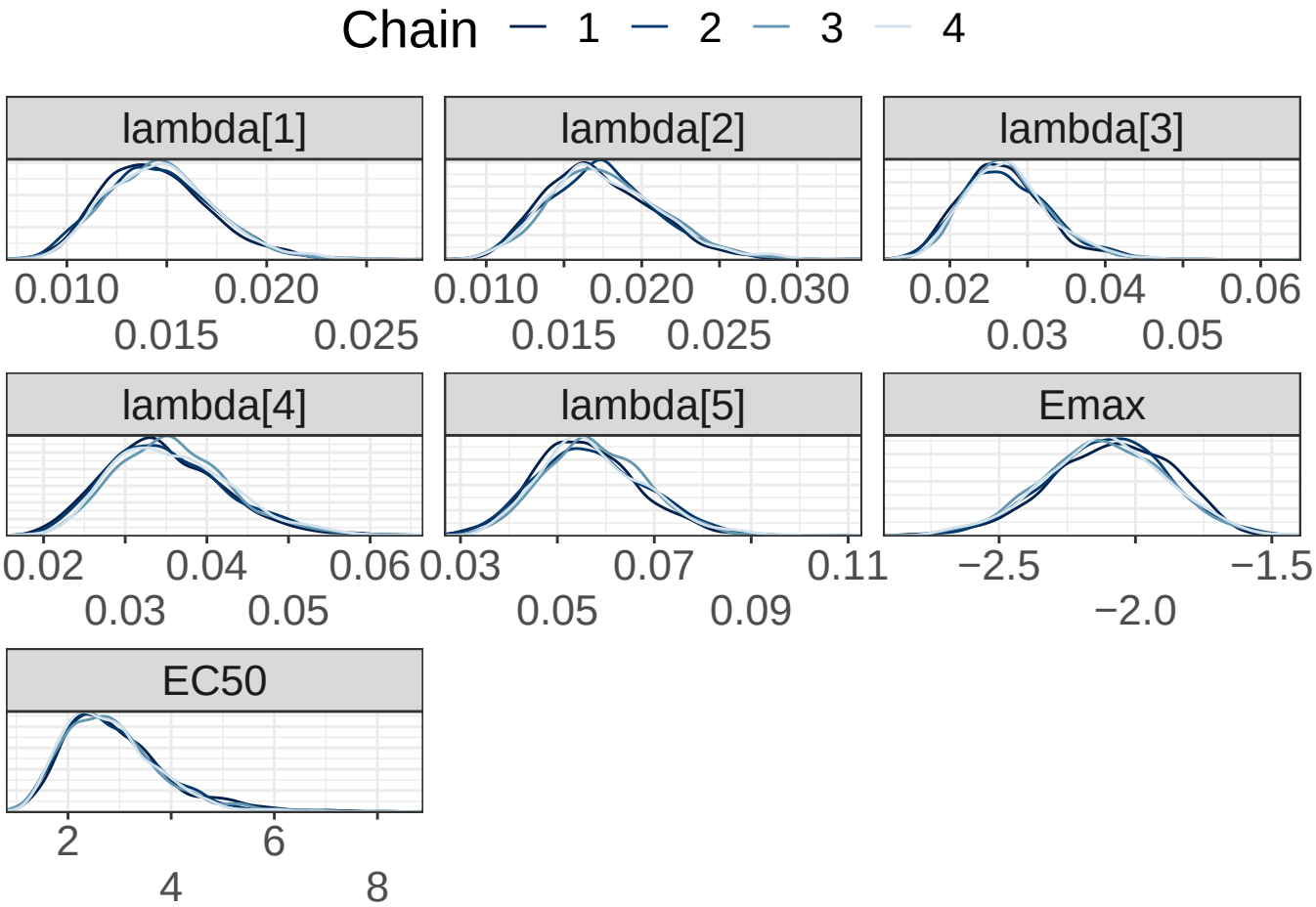
Fitted model: estimates

```
##   variable      mean   median   sd  mad        q5      q95 rhat ess_bulk ess_tail
##  lp__       -1564.52 -1564.16 1.91 1.75 -1568.21 -1562.05 1.00     1865     2636
##  lambda[1]      0.01     0.01 0.00 0.00     0.01     0.02 1.00     1677     2431
##  lambda[2]      0.02     0.02 0.00 0.00     0.01     0.02 1.00     1552     2348
##  lambda[3]      0.03     0.03 0.01 0.01     0.02     0.04 1.00     1569     2491
##  lambda[4]      0.04     0.03 0.01 0.01     0.02     0.05 1.00     1645     2514
##  lambda[5]      0.06     0.06 0.01 0.01     0.04     0.08 1.00     1667     2626
##  Emax          -2.10    -2.10 0.22 0.23    -2.47    -1.75 1.00     2148     2442
##  EC50           2.87     2.72 0.97 0.87     1.60     4.66 1.00     1790     2538
##
##  # showing 8 of 308 rows (change via 'max_rows' argument or 'cmdstanr_max_rows'
```

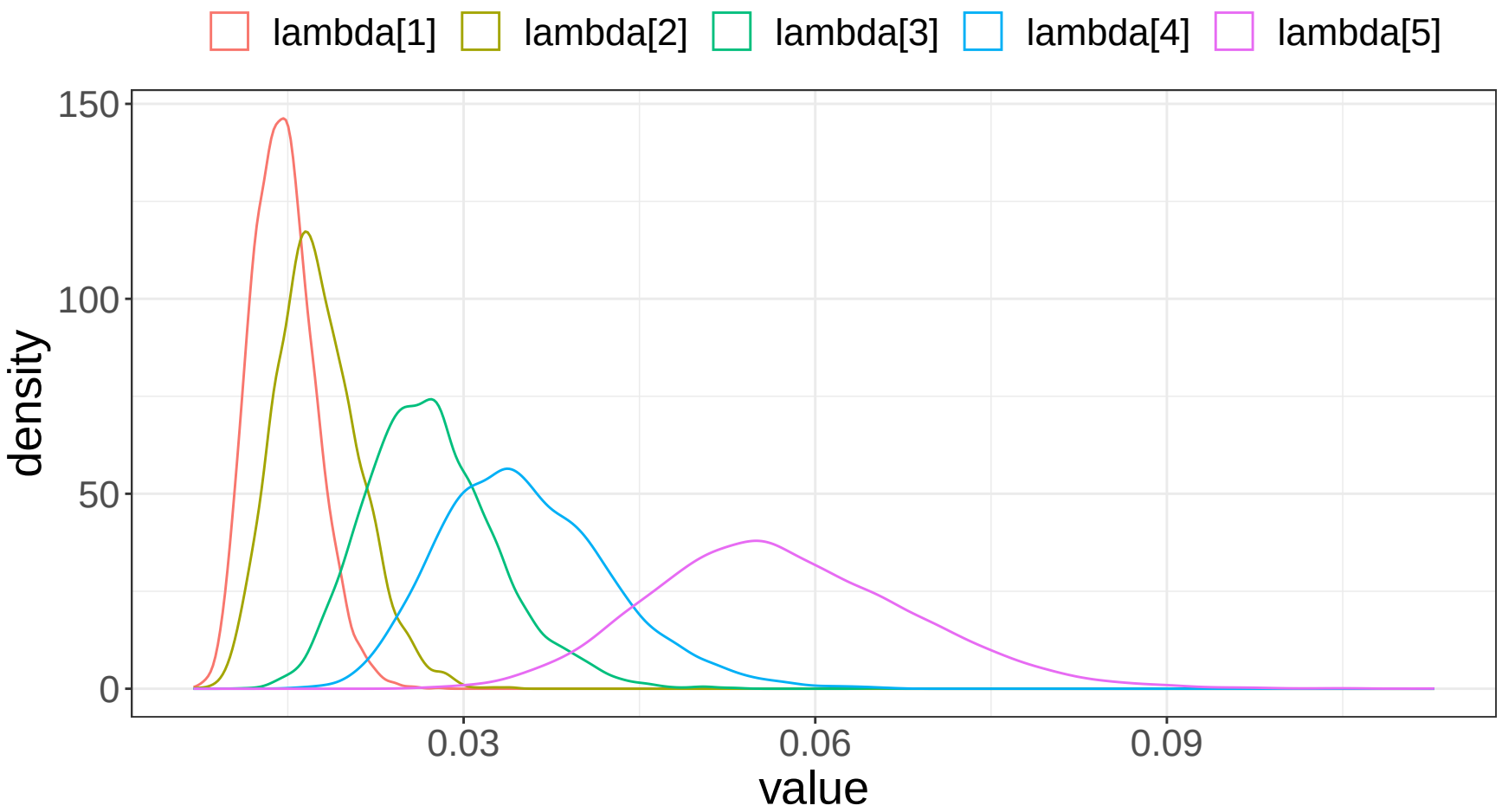
Fitted model: trace plots



Fitted model: density plots



Fitted model: compare lambdas



Visual predictive check

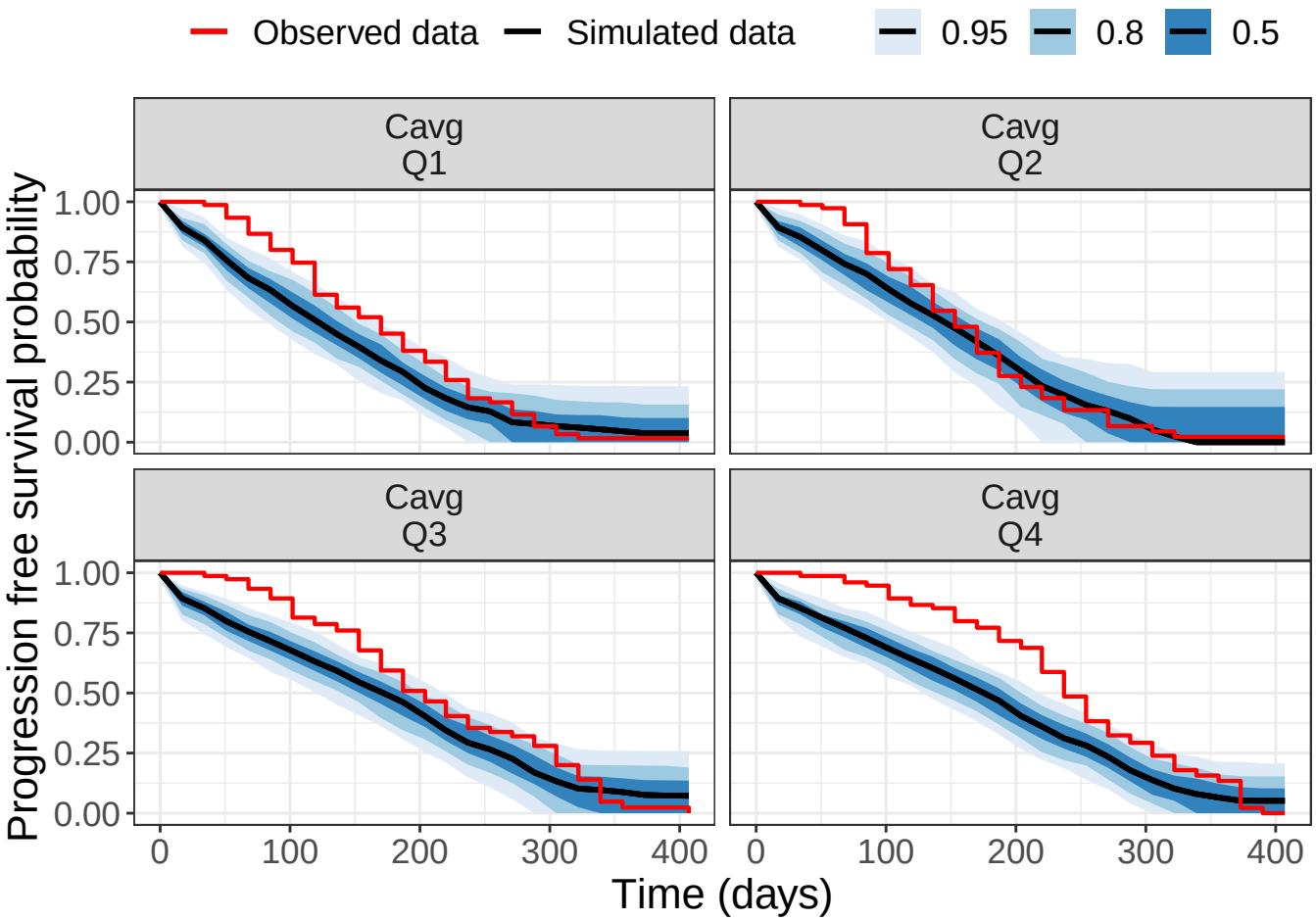


Table of Contents

Summary

- Exposure-response analyses play a pivotal role in evaluating dosing regimens
- Non-parametric and parametric survival analyses have their place in ER analysis, but semi-parametric models can offer greater flexibility
- Bayesian semi-parametric methods, particularly the piecewise exponential model, provide value in cases of time-varying exposure metrics

Table of Contents

Acknowledgements

- Jim Rogers
- Jonathan French
- Ramon Garcia
- Matthew Wiens

Statistics and Pharmacometrics Special Interest Group

`https://sxpsig.github.io/`

- Chartered by both the American Statistical Association and International Society of Pharmacometrics
- As of 2023, a Working Group of the ASA Biopharmaceutical Section

Table of Contents

References I

10 Betancourt, Michael. 2017. “A Conceptual Introduction to Hamiltonian Monte Carlo,” January. https://arxiv.org/abs/1701.02434.

Bradburn, M J, T G Clark, S B Love, and D G Altman. 2003. “Survival Analysis Part II: Multivariate Data Analysis–an Introduction to Concepts and Methods.” *Br. J. Cancer* 89 (3): 431–36.

Cox, D R. 1972. “Regression Models and Life-Tables.” *J. R. Stat. Soc.* 34 (2): 187–202.

French, Jonathan L, Neal Thomas, and Cunshan Wang. 2012. “Using Historical Data with Bayesian Methods in Early Clinical Trial Monitoring.” *Stat. Biopharm. Res.* 4 (4): 384–94.

Harrell, F E, Jr, R M Califf, D B Pryor, K L Lee, and R A Rosati. 1982. “Evaluating the Yield of Medical Tests.” *JAMA* 247 (18): 2543–46.

References II

Hoffman, Matthew D, and Andrew Gelman. 2014. “The No-U-turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo.” *J. Mach. Learn. Res.* 15 (1): 1593–623.

Holford, Nick. 2013. “A Time to Event Tutorial for Pharmacometricians.” *CPT Pharmacometrics Syst Pharmacol* 2 (May): e43.

Hosmer, D. W., S. Lemeshow, and S. May. 2011a. *Applied Survival Analysis: Regression Modeling of Time-to-Event Data*. Wiley Series in Probability and Statistics. Wiley. `https://books.google.com/books?id=IvvOopIqzWsC`.

———. 2011b. “Appendix 2: An Introduction to the Counting Process Approach to Survival Analysis.” In *Applied Survival Analysis*, 359–63. Hoboken, NJ, USA: John Wiley & Sons, Inc.

References III

Ibrahim, Joseph G, Ming-Hui Chen, and Debajyoti Sinha. n.d. *Bayesian Survival Analysis*. Springer New York.

Peng, Yaru, Zeneng Cheng, and Feifan Xie. 2021. "Evaluation of Pharmacokinetic Drug–Drug Interactions: A Review of the Mechanisms, in Vitro and in Silico Approaches." *Metabolites* 11 (January): 75. https://doi.org/10.3390/metabo11020075.

Qing, Yun, Peter F Thall, and Ying Yuan. 2023. "A Bayesian Piecewise Exponential Phase II Design for Monitoring a Time-to-Event Endpoint." *Pharm. Stat.* 22 (1): 34–44.

Royston, Patrick, and Mahesh K B Parmar. 2002. "Flexible Parametric Proportional-Hazards and Proportional-Odds Models for Censored Survival Data, with Application to Prognostic Modelling and Estimation of Treatment Effects." *Stat. Med.* 21 (15): 2175–97.

References IV

Ruiz-Garcia, Ana, Paul Baverel, Dean Bottino, Michael Dolton, Yan Feng, Ignacio González-García, Jaeyeon Kim, et al. 2023. "A Comprehensive Regulatory and Industry Review of Modeling and Simulation Practices in Oncology Clinical Drug Development." *J. Pharmacokinet. Pharmacodyn.* 50 (3): 147–72.

Wei, L J. 1992. "The Accelerated Failure Time Model: A Useful Alternative to the Cox Regression Model in Survival Analysis." *Stat. Med.* 11 (14-15): 1871–79.

Whitehead, John. 1980. "Fitting Cox's Regression Model to Survival Data Using GLIM." *J. R. Stat. Soc. Ser. C Appl. Stat.* 29 (3): 268–75.

Zecchin, Chiara, Ivelina Gueorguieva, Nathan H Enas, and Lena E Friberg. 2016. "Models for Change in Tumour Size, Appearance of New Lesions and Survival Probability in Patients with Advanced Epithelial Ovarian Cancer." *Br. J. Clin. Pharmacol.* 82 (3): 717–27.